ALIBABA CLOUD

Alibaba Cloud

Log Service

Application

Document Version: 20220712

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

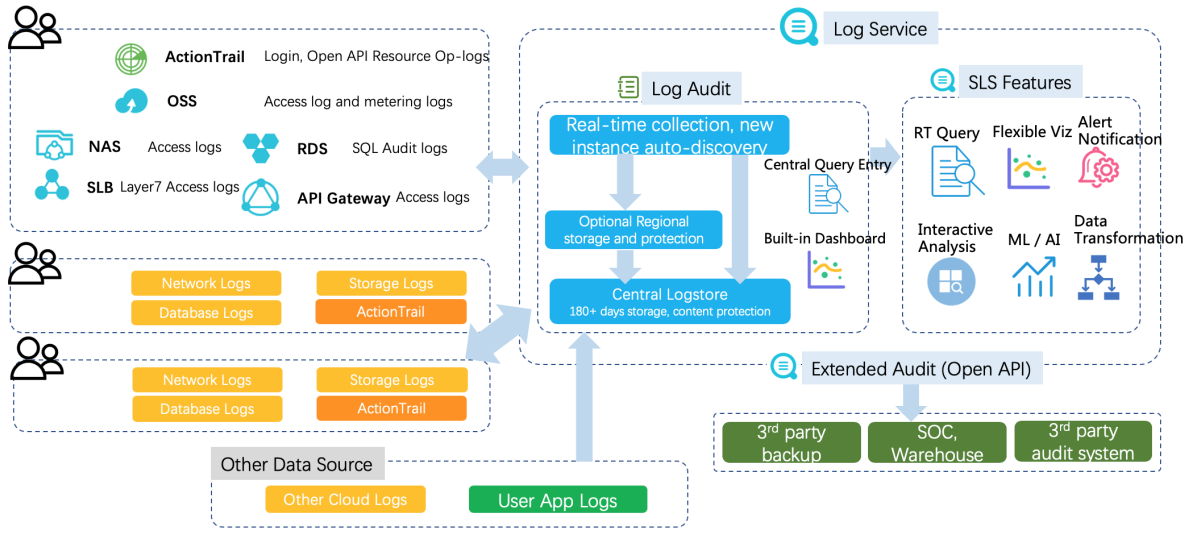
1.Log Audit Service

1.1. Overview of Log Audit Service

This topic describes the features, background information, scenarios, and benefits of Log Audit Service. This topic also describes the Alibaba Cloud services that are supported by Log Audit Service.

Features

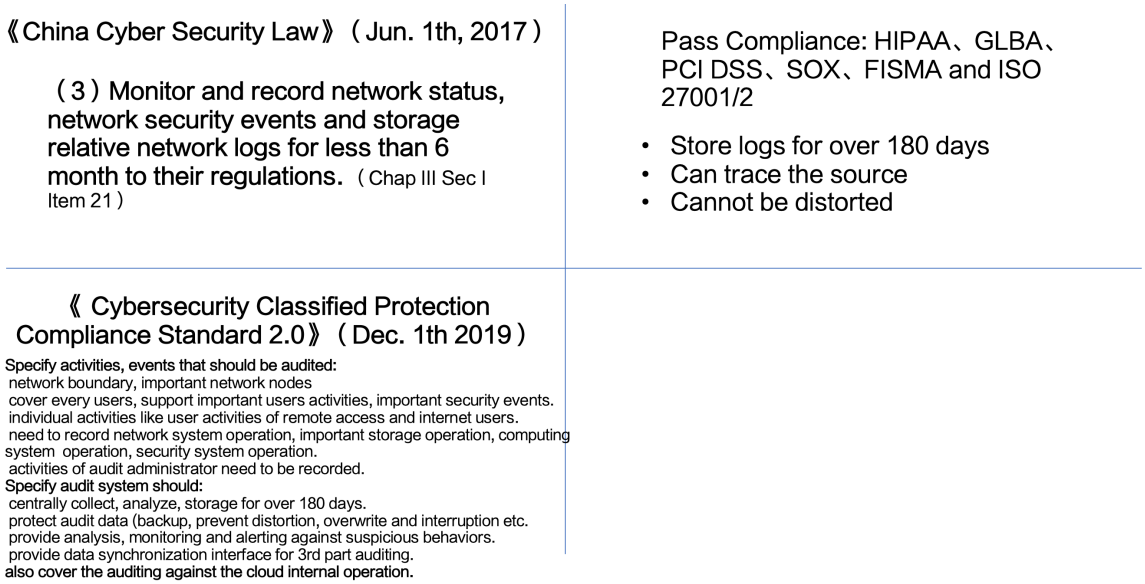
Log Audit Service supports all features of Log Service. Log Audit Service also supports automated and centralized log collection from cloud services across Alibaba Cloud accounts in real time. Then, you can audit the collected logs. In addition, Log Audit Service stores data required for audit and allows you to query and aggregate the data. You can use Log Audit Service to audit the logs that are collected from the following Alibaba Cloud services: ActionTrail, Container Service for Kubernetes (ACK), Object Storage Service (OSS), Apsara File Storage NAS (NAS), Server Load Balancer (SLB), Application Load Balancer (ALB), API Gateway, Virtual Private Cloud (VPC), ApsaraDB RDS, PolarDB-X 1.0, PolarDB, Web Application Firewall (WAF), Anti-DDoS, Cloud Firewall, and Security Center. You can also use Log Audit Service to audit the logs that are collected from third-party cloud services and self-managed security operations centers (SOCs).



Background information

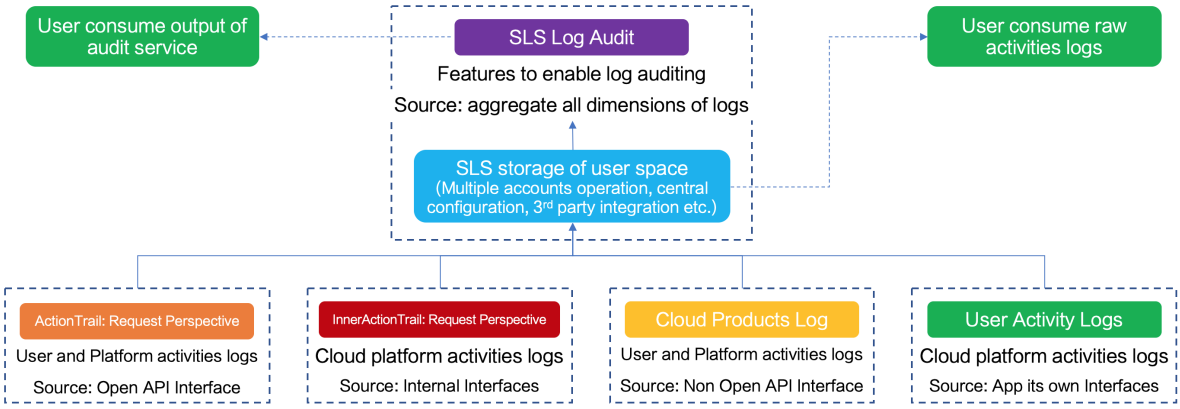
- Log audit is required by law.

Log audit is required by enterprises around the world to meet regulatory requirements. The Cybersecurity Law of the People's Republic of China came into effect in the Chinese mainland in 2017. In addition, the Multi-Level Protection Scheme (MLPS) 2.0 came into effect in December 2019.



- Log audit is the foundation for the data security compliance of enterprises.

  A large number of enterprises have compliance and audit teams that are capable of auditing device operations, network behavior, and logs. You can use Log Audit Service to consume raw logs, audit logs, and generate compliance audit reports. You can use your self-managed SOC or Alibaba Cloud Security Center to consume logs in Log Audit Service.



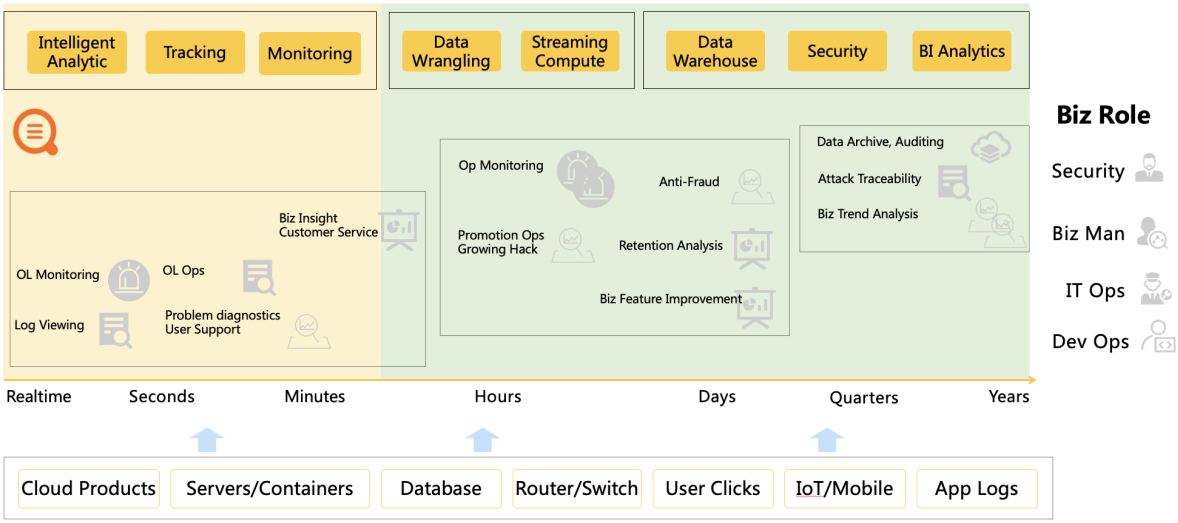
- Log audit is crucial for data security and protection.

  The M-Trends 2018 report published by FireEye stated that most enterprises, especially enterprises in Asia Pacific, are vulnerable to cybersecurity attacks. The global median dwell time was 101 days. In Asia Pacific, the median dwell time was 498 days. The dwell time indicates a period from when an attack occurs to when the attack is detected. To shorten the time, enterprises need reliable log data, durable storage, and audit services.

Scenarios

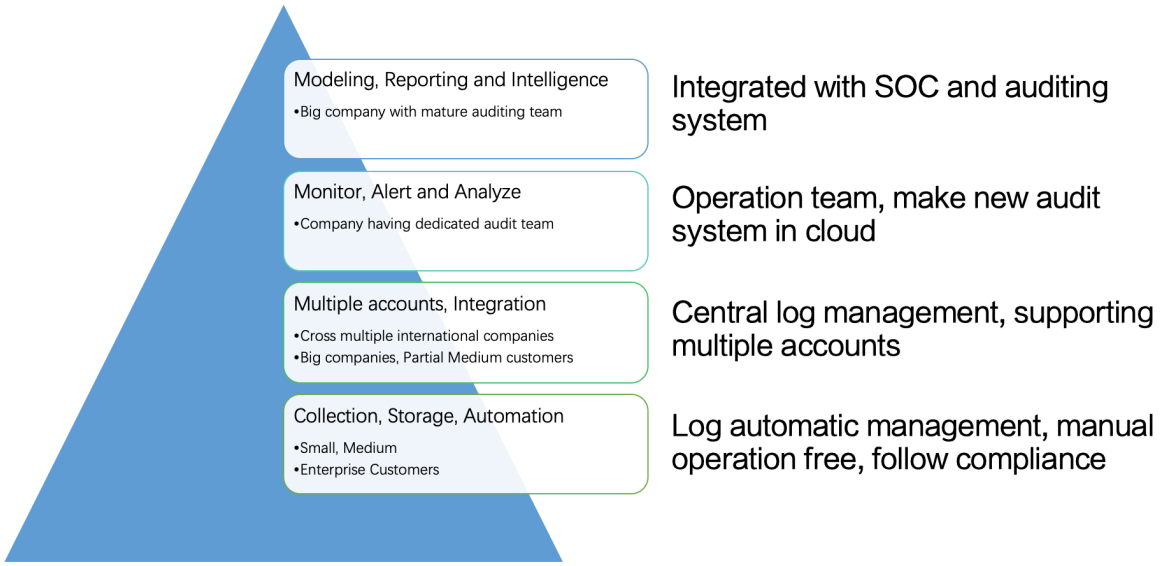
- Log Service-based audit

Log Service allows you to collect, cleanse, analyze, and visualize logs from end to end. You can also configure alerts for logs. You can use Log Service in DevOps, operations, security, and audit scenarios.



- Typical log audit

  The following requirements for log audit are classified into four levels.



  - Basic requirements: Most small and medium enterprises require automatic log collection and storage. These enterprises need to meet the basic requirements that are specified in MLPS 2.0 and implement automatic maintenance.
  - Intermediate requirements: Multinational enterprises, large enterprises, and some medium enterprises have multiple departments that use different Alibaba Cloud accounts and pay separate bills. However, logs required for audit must be automatically collected in a centralized manner. In addition to basic requirements, these enterprises need to collect logs and manage accounts in a centralized manner. In most cases, these enterprises have audit systems and need to synchronize their audit systems with Log Audit Service in real time.

- Advanced requirements: Large enterprises that have dedicated compliance and audit teams need to monitor logs, analyze logs, and configure alerts for logs. Some of the enterprises collect logs and send the logs to their audit systems for further processing. Other enterprises that want to build an audit system on the cloud can use the audit-related features provided by Log Service. The features include query, analysis, alerting, and visualization.
- Top requirements: Most large enterprises that have professional compliance and audit teams have self-managed SOCs or audit systems. These enterprises need to synchronize their SOCs or audit systems with Log Audit Service and manage data in a centralized manner.

Log Audit Service of Log Service meets all the four levels of requirements.

Benefits

- Centralized log collection
  - Log collection across accounts: You can collect logs from multiple Alibaba Cloud accounts to a project within one Alibaba Cloud account. You can configure multi-account collection in custom authentication mode or resource directory mode. The resource directory mode is recommended. For more information, see Configure multi-account collection.
  - Ease of use: You need to only configure collection policies once. Then, Log Audit Service collects logs in real time from Alibaba Cloud resources that belong to different accounts when new resources are detected. The new resources include newly created ApsaraDB RDS instances, SLB instances, and OSS buckets.
  - Centralized storage: Logs are collected and stored in the central project of a region. This way, you can query, analyze, and visualize the collected logs in a more efficient manner. You can also configure alerts for the logs and perform secondary development.
- Comprehensive audit
  - Log Audit Service supports all features of Log Service. For example, you can query, analyze, transform, visualize, and export logs, and configure alerts for logs. Log Audit Service also allows you to audit logs in a centralized manner.
  - You can use Log Audit Service together with Alibaba Cloud services, open source software, and third-party SOCs to create more value from data.

Supported Alibaba Cloud services

You can use Log Audit Service to audit the logs that are collected from the following Alibaba Cloud services: ActionTrail, ACK, OSS, NAS, SLB, ALB, API Gateway, VPC, ApsaraDB RDS, PolarDB-X 1.0, PolarDB, WAF, Cloud Firewall, Security Center, and Anti-DDoS. Logs that are collected from an Alibaba Cloud service are automatically stored in Logstores and Metricstores. Dashboards are automatically generated for the Logstores and Metricstores. The following table describes the details.

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
Action Trail	• RAM logon logs • Resource operation logs of Alibaba Cloud services • Logs of operations in OpenAPI Explorer	China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Japan (Tokyo), US (Silicon Valley), US (Virginia), Germany (Frankfurt), UK (London), India (Mumbai), and UAE (Dubai)	None	• Logstore actiontrail_log • Dashboard ◦ ActionTrail Audit Center ◦ ActionTrail Core Configuration Center ◦ ActionTrail Login Center
SLB	Layer 7 network logs of HTTP or HTTPS listeners	China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Chengdu), China (Hong Kong), Singapore (Singapore), Japan (Tokyo), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Philippines (Manila), India (Mumbai), UK (London), UAE (Dubai), Australia (Sydney), US (Silicon Valley), US (Virginia), and Germany (Frankfurt)	None	• Logstore slb_log • Dashboard ◦ SLB Audit Center ◦ SLB Access Center ◦ SLB Overall Data View

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
ALB	Layer 7 network logs of HTTP or HTTPS listeners	China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Ulanqab), China (Shenzhen), China (Guangzhou), China (Chengdu), China (Hong Kong), Japan (Tokyo), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Germany (Frankfurt), US (Silicon Valley), US (Virginia), and India (Mumbai)	None	• Logstore alb_log • Dashboard ◦ ALB Operation Center ◦ ALB Access Center
API Gateway	Access logs	All supported regions	None	• Logstore apigateway_log • Dashboard API Gateway Audit Center

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
VPC	Flow logs	China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Chengdu), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Japan (Tokyo), US (Silicon Valley), US (Virginia), UAE (Dubai), Germany (Frankfurt), India (Mumbai), and UK (London)	• After the flow log feature is enabled for a VPC or a vSwitch, the feature cannot capture information about ECS instances that belong to the following instance families in the VPC or vSwitch. The feature can capture information about only other ECS instances that meet the requirements. • The feature cannot be enabled for elastic network interfaces (ENIs) that are bound to ECS instances if the ECS instances belong to the following instance families. ecs.c1, ecs.c2, ecs.c4, ecs.ce4, ecs.cm4, ecs.d1, ecs.e3, ecs.e4, ecs.ga1, ecs.gn4, ecs.gn5, ecs.i1, ecs.m1, ecs.m2, ecs.mn4, ecs.n1, ecs.n2, ecs.n4, ecs.s1, ecs.s2, ecs.s3, ecs.se1, ecs.sn1, ecs.sn2, ecs.t1, and ecs.xn4	• Logstore vpc_log • Dashboard ◦ VPC Flow Log Overview ◦ VPC Flow Log Rejection Center ◦ VPC Flow Log Traffic Center
WAF	• Access logs • Attack logs	All supported regions	• Your WAF instance must be of the Business or Enterprise edition. • The **Log Service for WAF** feature must be enabled in the WAF console. For more information, see Enable the log analysis feature.	• Logstore waf_log • Dashboard ◦ WAF Audit Center ◦ WAF Security Center ◦ WAF Access Center

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
Security Center	• Seven types of host logs • Four types of network logs • Three types of security logs	China (Hangzhou) and Singapore (Singapore)	• Your Security Center must be of the Enterprise edition. • The **log analysis** feature must be enabled in the Security Center console. For more information, see Enable the log analysis feature.	• Logstore sas_log • Dashboard ◦ SAS Alarm Center ◦ SAS Connection Center ◦ SAS DNS Access Center ◦ SAS Baseline Center ◦ SAS Login Center ◦ SAS Process Center ◦ SAS Network Session Center ◦ SAS Vulnerability Center ◦ SAS Web Access Center
Cloud Firewall	Traffic logs of the Internet firewall and VPC firewalls	N/A	• Your Cloud Firewall must be of the Premium Edition or higher. • The **log analysis** feature must be enabled in the Cloud Firewall console. For more information, see Enable the log analysis feature.	• Logstore cloudfirewall_log • Dashboard Cloud Firewall Audit Center
Bastion host	Operation logs	All supported regions	Your Bastionhost must be of V3.2 or later.	• Logstore bastion_log • Dashboard None

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
OSS	• Resource operation logs • Data operation logs • Data access logs and metering logs • Deletion logs of expired files • CDN back-to-origin traffic logs	China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Chengdu), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Philippines (Manila), Japan (Tokyo), South Korea (Seoul), Thailand (Bangkok), India (Mumbai), Germany (Frankfurt), UAE (Dubai), UK (London), US (Virginia), and US (Silicon Valley)	None	• Logstore oss_log • Dashboard ◦ OSS Audit Center ◦ OSS Access Center ◦ OSS Operation Center ◦ OSS Performance Center ◦ OSS Overall Data View

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
Apsara DB RDS	• Audit logs of ApsaraDB RDS for MySQL instances • Slow query logs of ApsaraDB RDS for MySQL instances • Performance logs of ApsaraDB RDS for MySQL instances • Error logs of ApsaraDB RDS for MySQL instances	• Audit logs of ApsaraDB RDS for MySQL instances: all supported regions except China (Nanjing - Local Region), China (Heyuan), and Philippines (Manila) • Slow query logs of ApsaraDB RDS for MySQL instances: all supported regions except China (Nanjing - Local Region) and Philippines (Manila) • Performance logs of ApsaraDB RDS for MySQL instances: all supported regions except China (Nanjing - Local Region) and Philippines (Manila) • Error logs of ApsaraDB RDS for MySQL instances: all supported regions except China (Nanjing - Local Region) and Philippines (Manila)	• Audit logs ◦ ApsaraDB RDS for MySQL instances are supported, except those running the RDS Basic Edition. ◦ All ApsaraDB RDS for PostgreSQL and ApsaraDB RDS for SQL Server instances are supported. ◦ The SQL Explorer or SQL Audit feature must be enabled. The features are automatically enabled by Log Audit Service. • Slow query logs, performance logs, and error logs ApsaraDB RDS for MySQL instances are supported, except those running the RDS Basic Edition.	• Audit logs ◦ Logstore rds_log ◦ Dashboard ■ RDS Audit Center ■ RDS Security Center ■ RDS Performance Center ■ RDS Overall Data View • Slow query logs and error logs ◦ Logstore rds_log ◦ Dashboard None • Performance logs ◦ Metricstore rds_metrics ◦ Dashboard RDS Performance Monitor

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource

PolarDB	• Audit logs of PolarDB for MySQL clusters
	• Slow query logs of PolarDB for MySQL clusters
	• Performance logs of PolarDB for MySQL clusters
	• Error logs of PolarDB for MySQL clusters
		All supported regions	• Audit logs
			  ○ PolarDB for MySQL clusters are supported.
			  ○ The SQL Explorer or SQL Audit feature must be enabled. The features are automatically enabled by Log Audit Service.
			• Slow query logs, performance logs, and error logs
			Only PolarDB for MySQL clusters are supported.
				• Slow query logs, audit logs, and error logs
				  ○ Logstore
				polardb_log
				  ○ Dashboard
				None
				• Performance logs
				  ○ Metricstore
				polardb_metrics
				  ○ Dashboard
				PolarDB Performance Monitor

PolarDB-X 1.0	PolarDB-X 1.0 audit logs	China (Qingdao), China (Shenzhen), China (Shanghai), China (Beijing), China (Hangzhou), China (Zhangjiakou), China (Chengdu), and China (Hong Kong)	None	• Logstore
				drds_log
				• Dashboard
				  ○ DRDS Operation Center
				  ○ DRDS Security Center
				  ○ DRDS Performance Center

NAS	Access logs	All supported regions	None	• Logstore
				nas_log
				• Dashboard
				  ○ NAS Summary
				  ○ NAS Audit Center
				  ○ NAS Operation Center

			You must manually enable the log	

Alibaba Cloud service	Audited log	Supported region for the service	collection feature for Kubernetes logs. Prerequisite Note • You must use projects that are automatically created and are named in the k8s-log-{ClusterID} format. Projects that are manually created are not supported. • The collection of Kubernetes logs is based on the data transformation feature. When you collect Kubernetes logs, you are charged for the data transformation feature. For more information, see Billable items. • You cannot collect Kubernetes logs across accounts. • For more information about Kubernetes	Log Service resource

ACK	• Kubernetes audit logs
	• Kubernetes event centers
	• Ingress access logs
		China (Shanghai), China (Beijing), China (Hangzhou), China (Shenzhen), China (Hohhot), China (Zhangjiakou), China (Chengdu), and China (Hong Kong)		• Logstore
				  ◦ k8s_log
				  ◦ k8s_ingress_log
				• Dashboard
				  ◦ Kubernetes Audit Center Overview
				  ◦ Kubernetes Event Center
				  ◦ Kubernetes Resource Operation Overview
				  ◦ Ingress Overview
				  ◦ Ingress Access Center

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
			audit logs, see Collect log data from containers by using Log Service. • For more information about Kubernetes event centers, see Create and use an event center. • For more information about Ingress access logs, see Analyze and monitor the access log of nginx-ingress.	
Anti-DDoS	• Anti-DDoS Pro access logs • Anti-DDoS Premium access logs • Anti-DDoS Origin access logs	N/A	• Anti-DDoS Pro: The log analysis feature must be enabled in the Anti-DDoS Pro console. For more information, see Enable the log analysis feature. • Anti-DDoS Premium: The log analysis feature must be enabled in the Anti-DDoS Premium console. For more information, see Enable the log analysis feature. • Anti-DDoS Origin: The log analysis feature must be enabled in the Anti-DDoS Origin console. For more information, see Enable the mitigation analysis feature of Anti-DDoS Origin.	• Logstore ddos_log • Dashboard ◦ Anti-DDoS Premium Access Center ◦ Anti-DDoS Premium Operation Center ◦ Anti-DDoS Pro Access Center ◦ Anti-DDoS Pro Operation Center ◦ Anti-DDoS Origin Events Report ◦ Anti-DDoS Origin Mitigation Report

Alibaba Cloud service	Audited log	Supported region for the service	Prerequisite	Log Service resource
Cloud Service Bus (CSB) App Connect	Operation logs	N/A	None	• Logstore appconnect_log • Dashboard None

1.2. Usage notes

This topic describes the limits and billing of Log Audit Service.

Limits

- Storage methods and regions
  - Centralized storage

    Logs that are collected from multiple Alibaba Cloud accounts across different regions are stored in a central project of a central Alibaba Cloud account. A central project can reside in the following regions.

    > **Note** When you change the region of the central project within a central Alibaba Cloud account, Log Service creates a central project in the new region. The original project is not deleted.

    - Chinese mainland: China (Qingdao), China (Beijing), China (Hohhot), China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Hong Kong)
    - Outside the Chinese mainland: Singapore (Singapore), Japan (Tokyo), Germany (Frankfurt), and Indonesia (Jakarta)
  - Regional storage

    For Server Load Balancer (SLB), Application Load Balancer (ALB), Object Storage Service (OSS), and PolarDB-X 1.0, if the access logs are collected from multiple Alibaba Cloud accounts, Log Audit Service stores the collected logs in the projects that belong to the central Alibaba Cloud account and reside in the same regions as the cloud services. This also applies for the flow logs of Virtual Private Cloud (VPC). For example, if access logs are collected from an OSS bucket that resides in the China (Hangzhou) region, the access logs are stored in a project that also resides in the China (Hangzhou) region.
  - Synchronization to a central project

    For SLB, ALB, OSS, PolarDB-X 1.0, and VPC, if regional storage is used, you can synchronize logs from the Logstores of regional projects to the Logstores of a central project. This way, you can query, analyze, and visualize the logs in a more efficient manner. You can also configure alerts for the logs and perform secondary development.

    The synchronization process is based on the data transformation feature of Log Service.

- Resources
  - A central Alibaba Cloud account has only one functioning central project. The name of a central project is in the following format: slsaudit-center-*Alibaba Cloud account ID-Region specified for the central project*. Example: slsaudit-center-1234567890-cn-beijing. You cannot delete a central project in the Log Service console. If you want to delete a central project, you can use the Alibaba Cloud command-line interface (CLI) or call API operations.
  - For SLB, ALB, OSS, PolarDB-X 1.0, and VPC, logs can be stored in multiple regional projects. The name of a regional project is in the following format: slsaudit-region-*Alibaba Cloud account ID-Source region for collection*. Example: slsaudit-region-1234567890-cn-beijing. You cannot delete a regional project in the Log Service console. If you want to delete a regional project, you can use the Alibaba Cloud CLI or call API operations.
  - If you enable log collection for a cloud service, Log Audit Service creates a dedicated Logstore. You can manage a dedicated Logstore in the same way that you manage other Logstores. A dedicated Logstore has the following limits:
    - To prevent data tampering, Log Service allows only the specified service to write logs to the dedicated Logstore. You cannot modify or delete indexes in the Logstore.
    - You can modify the retention period of logs or delete the dedicated Logstore only on the Global Configurations page of Log Audit Service or by calling API operations.
    - For SLB, ALB, OSS, PolarDB-X 1.0, and VPC, if **Synchronization to Central Project** is enabled, data transformation jobs are generated in the regional projects.
      - The data transformation job that is generated for OSS logs is named Internal Job: SLS Audit Service Data Sync for OSS Access. The data transformation job that is generated for SLB logs is named Internal Job: SLS Audit Service Data Sync for SLB. The data transformation job that is generated for ALB logs is named Internal Job: SLS Audit Service Data Sync for ALB. The data transformation job that is generated for DRDS logs is named Internal Job: SLS Audit Service Data Sync for DRDS. The data transformation job that is generated for VPC logs is named Internal Job: SLS Audit Service Data Sync for VPC.
      - You can stop the data transformation jobs only on the Global Configurations page of Log Audit Service or by calling API operations.
      - If you turn on **Synchronization to Central Project**, the logs in the Logstores of the regional projects are synchronized to the dedicated Logstores of the central project. You can no longer manage the Logstores of the regional projects. However, you can perform operations such as queries on the Logstores of the central project.
- Data retention periods in days
  - In Log Audit Service, the audit logs, slow query logs, and error logs of ApsaraDB RDS for MySQL instances are stored in the same Logstore, which is named rds_log. If log collection is enabled for all types of logs but the data retention periods are different, the largest value of the data retention periods is used.
  - In Log Audit Service, the audit logs, slow query logs, and error logs of PolarDB for MySQL clusters are stored in the same Logstore, which is named polardb_log. If log collection is enabled for all types of logs but the data retention periods are different, the largest value of the data retention periods is used.
  - In Log Audit Service, the traffic logs of the Internet firewall and VPC firewalls in Cloud Firewall are stored in the same Logstore, which is named cloudfirewall_log. If log collection is enabled for both types of traffic logs but the data retention periods are different, the larger value of the data retention periods is used.

- In Log Audit Service, the access logs of Anti-DDoS Pro, Anti-DDoS Premium, and Anti-DDoS Origin are stored in the same Logstore, which is named ddos_log. If log collection is enabled for all types of access logs but the data retention periods are different, the largest value of the data retention periods is used.
- In Log Audit Service, the audit logs of Kubernetes clusters and the events of K8s Event Center are stored in the same Logstore, which is named k8s_log. If log collection is enabled for the audit logs and events but the data retention periods are different, the larger value of the data retention periods is used.

> **Note** The preceding list describes the types of logs whose data retention periods are affected by each other. If you enable both log collection and hot and cold-tiered storage for these types of logs, the hot retention period of the logs is the largest value of the hot retention periods for these types of logs. If you enable log collection for all these types of logs but enable hot and cold-tiered storage only for some types of logs, hot and cold-tiered storage is automatically disabled for all the logs.
>
> For example, if you enable log collection and hot and cold-tiered storage for the audit logs and error logs of ApsaraDB RDS for MySQL instances, the larger value of the hot retention periods for the audit logs and error logs is used. If you enable log collection for the audit logs and error logs of ApsaraDB RDS for MySQL instances but enable hot and cold-tiered storage only for the audit logs, hot and cold-tiered storage is disabled for the rds_log Logstore in which the logs are stored.

- Hot and cold-tiered storage

  The dedicated Logstores of Log Audit Service support the hot and cold-tiered storage feature. Cold storage costs lower than hot storage but reduces query and analysis performance. However, the performance of other operations, such as alerting, visualization, transformation, and shipping, is not reduced. For more information, see Enable hot and cold-tiered storage for a Logstore.

  > **Note** Log Audit Service allows you to enable the hot and cold-tiered storage feature in the following regions: China (Qingdao), China (Beijing), China (Hohhot), China (Hangzhou), China (Shanghai), and China (Shenzhen).

  You can enable the hot and cold-tiered storage feature on the **Global Configurations** page of Log Audit Service. The hot data retention period must be greater than or equal to 30 days but cannot exceed the current data retention period. For example, if the data retention period of a central project is 180 days and the hot data retention period is 30 days, hot data is moved to the cold storage after 30 days.

Billing

- Log Service

  You must activate Log Service and enable Log Audit Service for the central Alibaba Cloud account that is used to collect logs from other Alibaba Cloud accounts. You do not need to activate Log Service for the other Alibaba Cloud accounts. However, if the cloud services within the other Alibaba Cloud accounts rely on Log Service, you must activate Log Service for these accounts. No fees for Log Service are generated in these accounts. When you use Log Audit Service, you are charged for the data storage, read and write traffic, and data transformation based on the pay-as-you-go billing method. For more information, see Billable items.

> **Notice**
>
> - For SLB, ALB, OSS, PolarDB-X 1.0, and Container Service for Kubernetes (ACK), if Synchronization to Central Project is enabled, the collected logs are synchronized based on the data transformation feature. You are charged for data transformation and cross-network traffic based on the pay-as-you-go billing method. For more information, see Billable items.
> - You can use Log Audit Service or a common collection method to collect logs. You are charged when you use any of the two methods. If you use both methods to collect logs, Log Service stores two copies of data. You can use the two copies of data in different scenarios.
>   - Log Audit Service: This application supports automated and centralized log collection from cloud services across multiple Alibaba Cloud accounts in real time. The collected logs are used for compliance and auditing.
>   - Common method: Logs are collected by region and separately managed. The collected logs are used for log analysis. For more information, see Alibaba Cloud service logs.

You can use free resource quotas or purchase resource plans to offset your fees.

- Cloud services

  After you enable Log Audit Service in the Log Service console and enable log collection for cloud services, you may be charged additional fees. The fees are included in the bills for the cloud services. The following table describes the cloud services that may generate additional fees.

Cloud service	Additional fee
Web Application Firewall (WAF)	You are charged for the **Log Service for WAF** feature that is purchased in the WAF console. For more information about the feature fees, see Billing.
Security Center (SAS)	You are charged for the **log analysis** feature that is enabled in the Security Center console. For more information about the feature fees, see Billing.
Cloud Firewall	You are charged for the **log analysis** feature that is enabled in the Cloud Firewall console. For more information about the feature fees, see Billing.
	After you enable log collection for ApsaraDB RDS, the SQL Explorer or SQL Audit feature is automatically enabled on the ApsaraDB RDS instances that meet the requirements. All editions of ApsaraDB RDS for PostgreSQL and ApsaraDB RDS for SQL Server are supported. Only the Basic Edition of ApsaraDB RDS for MySQL is not supported. You are charged for the SQL Explorer or SQL Audit feature. For more information about the feature fees, see Billable items, billing methods, and pricing.

Cloud service	Additional fee
ApsaraDB RDS	**Note** ◦ If you have enabled SQL Explorer Trial Edition for your ApsaraDB RDS instance, Log Audit Service automatically disables SQL Explorer Trial Edition and enables the SQL Explorer feature after log collection is enabled. ◦ By default, the logs that are generated by the SQL Explorer feature are stored for 30 days. If you want to change the storage duration, you must perform the operation in the ApsaraDB RDS console. For more information, see Modify the retention period of SQL audit logs. The storage duration is independent of the data retention period in Log Audit Service that is specified for the audit logs of your ApsaraDB RDS instance. The storage duration and data retention period do not affect each other. If the storage duration that you specify in the ApsaraDB RDS console is less than 30 days, the logs cannot be shipped to Log Service. Log Audit Service automatically changes the duration to 30 days. ◦ If you have stopped collecting the audit logs of your ApsaraDB RDS instance and want to disable the SQL Explorer feature, you must disable the feature in the ApsaraDB RDS console. For more information, see Disable the SQL Explorer feature.

Cloud service	Additional fee
PolarDB	After you enable log collection for PolarDB, the SQL Explorer or SQL Audit feature is automatically enabled on the PolarDB clusters that meet the requirements. Only PolarDB for MySQL clusters are supported. You are charged for the SQL Explorer or SQL Audit feature. For more information about the feature fees, see Billable items. **Note** ◦ If you have enabled the trail edition of the SQL Explorer feature for your PolarDB for MySQL cluster, Log Audit Service automatically disables the trial edition and enables the official edition of the SQL Explorer feature after log collection is enabled. ◦ By default, the logs that are generated by the SQL Explorer feature are stored for 30 days. If you want to change the storage duration, you must perform the operation in the ApsaraDB PolarDB console. For more information, see Change the retention period of SQL logs. The storage duration is independent of the data retention period in Log Audit Service that is specified for the audit logs of your PolarDB for MySQL cluster. The storage duration and data retention period do not affect each other. If the storage duration that you specify in the ApsaraDB PolarDB console is less than 30 days, the logs cannot be shipped to Log Service. Log Audit Service automatically changes the duration to 30 days. ◦ If you have stopped collecting the audit logs of your PolarDB for MySQL cluster and want to disable the SQL Explorer feature, you must disable the feature in the ApsaraDB PolarDB console. For more information, see Disable the SQL Explorer feature.

Cloud service	Additional fee
Anti-DDoS	You are charged for the **log analysis** feature that is purchased in the Anti-DDoS Pro console. For more information about the feature fees, see Overview.
VPC	You are charged for network log extraction based on the amount of log data that is extracted. For more information, see Overview of the flow log feature.

1.3. Enable log collection

Log Audit Service allows you to enable the log collection feature with a few clicks. This topic describes how to enable the log collection feature and perform related operations.

Prerequisites

- An Alibaba Cloud account is created.

  We recommend that you use a RAM user of the Alibaba Cloud account to enable log collection. The RAM user must be granted the read permissions on RAM resources and the read and write permissions on Log Service resources. To grant the required permissions to the RAM user, you can attach the AliyunRAMReadOnlyAccess and AliyunLogFullAccess policies to the RAM user.

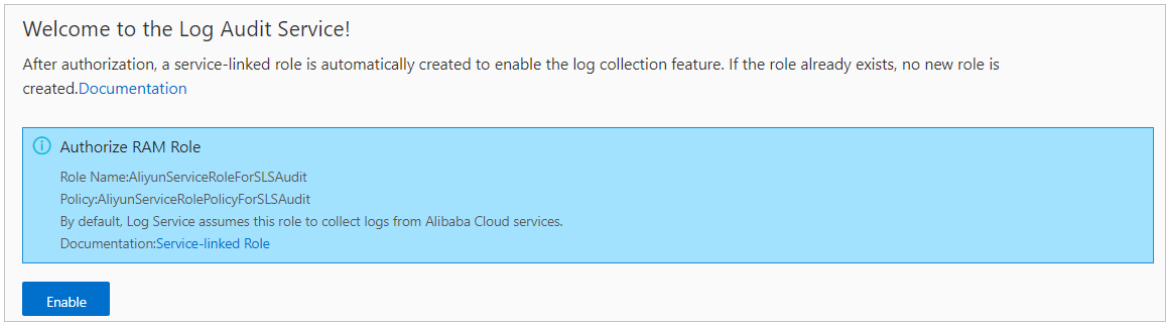
- The required features are enabled for the Alibaba Cloud services from which you want to collect logs. For more information, see Supported Alibaba Cloud services.

Initially configure Log Audit Service

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. Complete authorization by following the on-screen instructions.

   After you complete the authorization, Log Audit Service assumes the AliyunServiceRoleForSLSAudit service-linked role to collect logs from Alibaba Cloud services. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

   > **Notice**
   > - The account that you use to complete the authorization must have the permissions specified by the AliyunRamFullAccess policy.
   > - You need to complete the authorization only once.

Enable log collection

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Access to Cloud Products > Global Configurations**.
4. In the **Region of the Central Project** drop-down list, select the region of the project in which you want to centrally store the collected logs.
   - Chinese mainland: China (Qingdao), China (Beijing), China (Hohhot), China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Hong Kong)
   - Outside the Chinese mainland: Singapore (Singapore), Japan (Tokyo), Germany (Frankfurt), and Indonesia (Jakarta)
5. In the Cloud Products column, find the service for which you want to enable log collection and specify the retention period of logs.

   If you want to collect Layer 7 access logs from Server Load Balancer (SLB), Layer 7 access logs from Application Load Balancer (ALB), access logs from Object Storage Service (OSS), and audit logs from PolarDB-X 1.0, you can turn on the corresponding switches in the **Synchronization to Central Project** column. After you turn on a switch in the **Synchronization to Central Project** column, Log Service stores data in the regional project of the service only for the recommended period of time. The regional project of the service is used only as temporary storage.
6. Click **Save**.
   After the configuration is complete, wait for approximately 2 minutes to view the collection status of logs on the **Access to Cloud Products > Status Dashboard** page. If an exception occurs, modify the configurations by following the on-screen instructions. For more information, see Enable log collection.

What to do next

Stop log collection

If you no longer need to collect logs from an Alibaba Cloud service but you want to retain the collected logs, perform the following steps. Log Service deletes logs after the retention period of the logs elapses.

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Access to Cloud Products > Global Configurations**.
4. On the **Global Configurations** page, click **Modify** in the upper-right corner.
5. Find the Alibaba Cloud service and turn off the switch in the Audit-Related Logs column. Then, click **OK**.

Delete audit resources

If you want to delete Log Audit Service resources, such as projects, Logstores, dashboards, and alerts, perform the following steps:

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Access to Cloud Products > Global Configurations**.
4. On the **Global Configurations** page, click **Delete Audit Resources** in the upper-right corner.
5. In the **Delete All Resources of Log Audit Service** dialog box, click **Disable Log Collection for**

Cloud Services.

6. In the **Confirm** message, click **OK**.
7. In the **Delete All Resources of Log Audit Service** dialog box, copy commands based on your business requirements.

   If you want to delete all resources, copy all commands. If you want to delete specific resources, copy the required commands. Sample commands:

   > **Note**
   > - Run commands in sequence to delete a regional project before a central project.
   > - Before you delete a project, wait for 1 to 2 minutes to make sure that log collection is disabled for all Alibaba Cloud services.

   - Sample command to delete a regional project

     ```
     aliyunlog log delete_project --project_name=slsaudit-region-12****34-cn-huhehaote --r
     egion-endpoint=cn-huhehaote.log.aliyuncs.com
     ```

   - Sample command to delete a central project

     ```
     aliyunlog log delete_project --project_name=slsaudit-center-12****34-cn-huhehaote --r
     egion-endpoint=cn-huhehaote.log.aliyuncs.com
     ```

   In the preceding commands, *12****34* specifies the ID of the Alibaba Cloud account, and *cn-huhehaote* specifies the region of the projects. region-endpoint specifies the access endpoint of the projects. For more information, see Endpoints.

8. In the top navigation bar, click the icon.
9. On the **cloudshell** tab, run the commands that you copied.

   The system runs the commands one by one to delete audit resources.

1.4. Configure multi-account collection

Log Audit Service allows you to collect logs across Alibaba Cloud accounts. You can collect logs from the cloud services of other Alibaba Cloud accounts and store the logs in the Logstores within your Alibaba Cloud account. You cannot collect Kubernetes logs across Alibaba Cloud accounts. This topic describes how to configure multi-account collection.

Prerequisites

- Resource directory mode (recommended)
  - A member is created or invited. For more information, see Create a member or Invite an Alibaba Cloud account to join a resource directory.
  - The log collection feature is enabled. For more information, see Enable log collection.
- Custom authentication mode

  The log collection feature is enabled. For more information, see Enable log collection.

Context

Log Audit Service allows you to collect logs from cloud services across Alibaba Cloud accounts. You can configure multi-account collection in resource directory mode or custom authentication mode. Log Audit Service is integrated with Resource Directory to support the resource directory mode. You can invite other Alibaba Cloud accounts in your enterprise to join your resource directory by using a management account or a delegated administrator account. Then, you can collect logs from cloud services that belong to these Alibaba Cloud accounts. For more information about Resource Directory, see What is Resource Management?

For more information about the limits on the resource directory mode for multi-account collection, see Limits on resource directories.

Mode	Method	Description
Resource directory mode	All members	Log Audit Service automatically adds all members in your resource directory to the collection list and collects logs from the cloud services that belong to the members and have the log collection feature enabled. • After a member is added to your resource directory, the member is automatically included in the collection list. • After a member is removed from your resource directory, the member is automatically removed from the collection list.
	Custom	You can manually specify and add members to the collection list. This way, Log Audit Service collects logs from the cloud services that belong to the members and have the log collection feature enabled. • After a member is added to your resource directory, the member is not automatically included in the collection list. • After a member is removed from your resource directory, the member is automatically removed from the collection list if the member is still in the list.
Custom authentication mode	AccessKey pair-based authorization	You can configure multi-account collection by using the AccessKey pair of an Alibaba Cloud account or a RAM user.
	Manual authorization	You must complete manual authorization before you can configure multi-account collection. **Notice** Manual authorization is prone to errors, which may cause Log Audit Service to be unavailable. This method is not recommended.

Notice

- After you configure multi-account collection in resource directory mode, you cannot switch to the custom authentication mode. If you want to switch to the custom authentication mode, you must clear the existing configurations.
- If you reconfigure multi-account collection in resource directory mode after you configure multi-account collection in custom authentication mode, the configurations for the resource directory mode overwrite those for the custom authentication mode.

Resource directory mode (recommended)

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Multi-Account Configurations > Global Configurations**.
4. On the **Resource Directory Mode** tab, click **Modify**.
5. In the **AddAccount** panel, select the accounts that you want to invite and click **Confirm**.

   In resource directory mode, the All Members and Custom modes are supported.

   - All Members: Log Audit Service automatically adds all members in your resource directory to the collection list and collects logs from the cloud services that belong to the members and have the log collection feature enabled.
   - Custom: You can manually specify and add members to the collection list. This way, Log Audit Service collects logs from the cloud services that belong to the members and have the log collection feature enabled.

   After the configuration is complete, wait for approximately 2 minutes to view the collection status of logs on the **Access to Cloud Products > Status Dashboard** page. If an exception occurs, modify the configurations by following the on-screen instructions. For more information, see Enable log collection.

Custom authentication mode

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Multi-Account Configurations > Global Configurations**.
4. On the **Custom Authentication Mode** tab, click **Modify**.
5. Specify the account that you want to invite and click **OK**.

   In custom authentication mode, the AccessKey Pair-Based Authorization and Manual Authorization modes are supported.

   - AccessKey Pair-Based Authorization: Enter the ID of the Alibaba Cloud account that you want to invite and the required AccessKey pair. The AccessKey pair is for temporary use and is not saved.

     If you enter the AccessKey pair of a RAM user, the RAM user must have the read and write permissions on RAM resources. To grant the permissions, you can attach the AliyunRAMFullAccess policy to the RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.

- Manual Authorization: Enter the ID of the Alibaba Cloud account that you want to invite. You can enter multiple IDs. You must separate multiple IDs with line breaks, commas (,), spaces, or vertical bars (|). For more information about how to grant permissions to an account, see Use a custom policy to authorize Log Service to collect and synchronize logs.

After the configuration is complete, wait for approximately 2 minutes to view the collection status of logs on the **Access to Cloud Products > Status Dashboard** page. If an exception occurs, modify the configurations by following the on-screen instructions. For more information, see Enable log collection.

1.5. Configure log collection policies

The Log Audit Service application allows you to collect logs from Alibaba Cloud services across multiple accounts and store the logs in a centralized manner. If the log audit feature is enabled for an Alibaba Cloud service, Log Service collects all logs that meet specified conditions from the service by default. You can configure log collection policies to specify the accounts, regions, and instances from which logs are collected. This way, you can collect logs at a fine-grained level. This topic describes how to configure log collection policies.

Supported Alibaba Cloud services

You can configure log collection policies for ApsaraDB RDS, PolarDB-X 1.0, PolarDB, Server Load Balancer (SLB), Application Load Balancer (ALB), Virtual Private Cloud (VPC), and Container Service for Kubernetes (ACK). The following table provides the details of the services.

Alibaba Cloud service	Log source	Property	Description
RDS	ApsaraDB RDS instance	Account: account.id	The ID of the Alibaba Cloud account to which the ApsaraDB RDS instance belongs.
		Region: region	The ID of the region where the ApsaraDB RDS instance resides. Example: cn-shanghai.
		Instance ID: instance.id	The ID of the ApsaraDB RDS instance.
		Instance Name: instance.name	The name of the ApsaraDB RDS instance.
		DB Type: instance.db_type	The type of the databases that are created on the ApsaraDB RDS instance. Valid values: mysql, pgsql, and mssql.
		DB Version: instance.db_version	The version of the database engine. Example: 8.0.

Alibaba Cloud service	Log source	Property	Description
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
PolarDB	PolarDB cluster	Account: account.id	The ID of the Alibaba Cloud account to which the PolarDB cluster belongs.
		Region: region	The ID of the region where the PolarDB cluster resides. Example: cn-shanghai.
		Cluster ID: cluster.id	The ID of the PolarDB cluster.
		Cluster Name: cluster.name	The name of the PolarDB cluster.
		DB Type Compatible with Cluster: cluster.db_type	The database type that is supported by the PolarDB cluster. Valid value: MySQL.
		DB Version Compatible with Cluster: cluster.db_version	The version of the database engine. Valid values: 5.6, 5.7, and 8.0.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
PolarDB-X 1.0	PolarDB-X 1.0 instance	Account: account.id	The ID of the Alibaba Cloud account to which the PolarDB-X 1.0 instance belongs.
		Region: region	The ID of the region where the PolarDB-X 1.0 instance resides. Example: cn-shanghai.
		Instance ID: instance.id	The ID of the PolarDB-X 1.0 instance.
		Instance Name: instance.name	The name of the PolarDB-X 1.0 instance.
		Account: account.id	The ID of the Alibaba Cloud account to which the SLB instance belongs.

Alibaba Cloud service	Log source	Property	Description
SLB	SLB instance	Region: region	The ID of the region where the SLB instance resides. Example: cn-shanghai.
		Instance ID: instance.id	The ID of the SLB instance.
		Instance Name: instance.name	The name of the SLB instance.
		Network Type: instance.network_type	The network type of the SLB instance. Valid values: vpc and classic.
		VPC ID: instance.vpc_id	The ID of the VPC where the SLB instance resides.
		Address Type: instance.address_type	The address type of the SLB instance. Valid values: intranet and internet.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
ALB	ALB instance	Account: account.id	The ID of the Alibaba Cloud account to which the ALB instance belongs.
		Region: region	The ID of the region where the ALB instance resides. Example: cn-shanghai.
		Instance ID: instance.id	The ID of the ALB instance.
		Instance Name: instance.name	The name of the ALB instance.
		VPC ID: instance.vpc_id	The ID of the VPC where the ALB instance resides.
		Address Type: instance.address_type	The address type of the ALB instance. Valid values: Intranet and Internet.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.

Alibaba Cloud service	Log source	Property	Description
VPC	VPC	Account: account.id	The ID of the Alibaba Cloud account to which the VPC belongs.
		Region: region	The ID of the region where the VPC resides.
		Instance ID: instance.id	The ID of the VPC.
		Instance Name: instance.name	The name of the VPC.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
ACK (Kubernetes audit log)	Kubernetes cluster	Region: region	The ID of the region where the Kubernetes cluster resides. Example: cn-shanghai.
		Cluster ID: cluster.id	The ID of the Kubernetes cluster.
		Cluster Name: cluster.name	The name of the Kubernetes cluster.
		Cluster Type: cluster.type	The type of the Kubernetes cluster. Valid values: Kubernetes, ManagedKubernetes, and ASK.
		Network Type: cluster.network_mode	The network type of the Kubernetes cluster. Valid values: vpc and classic.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
		Region: region	The ID of the region where the Kubernetes cluster resides. Example: cn-shanghai.
		Cluster ID: cluster.id	The ID of the Kubernetes cluster.
		Cluster Name: cluster.name	The name of the Kubernetes cluster.

Alibaba Cloud service	Log source	Property	Description
ACK (Kubernetes event center)	Kubernetes cluster	Cluster Type: cluster.type	The type of the Kubernetes cluster. Valid values: Kubernetes, ManagedKubernetes, and ASK.
		Network Type: cluster.network_mode	The network type of the Kubernetes cluster. Valid values: vpc and classic.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
ACK (Ingress access log)	Kubernetes cluster	Region: region	The ID of the region where the Kubernetes cluster resides. Example: cn-shanghai.
		Cluster ID: cluster.id	The ID of the Kubernetes cluster.
		Cluster Name: cluster.name	The name of the Kubernetes cluster.
		Cluster Type: cluster.type	The type of the Kubernetes cluster. Valid values: Kubernetes, ManagedKubernetes, and ASK.
		Network Type: cluster.network_mode	The network type of the Kubernetes cluster. Valid values: vpc and classic.
		Tag: tag.*	The custom tag. You can replace the asterisk (*) in the tag.* property with a custom tag name.
		Log: log.*	The content of the log.

Configure a log collection policy

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. Choose **Access to Cloud Products > Global Configurations**. In the upper-right corner of the page that appears, click **Modify**.
4. Find the Alibaba Cloud service for which you want to configure a log collection policy and click **Collection Policy**.

5. Configure a log collection policy.

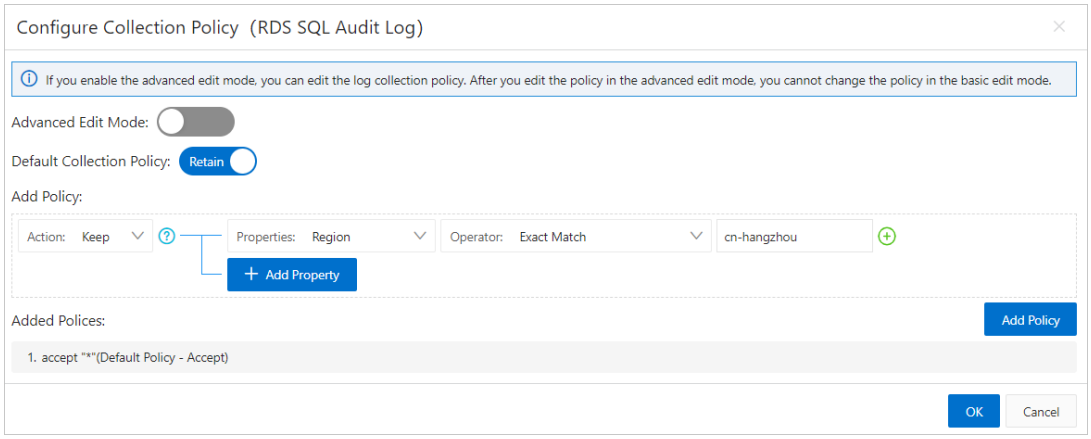
   You can configure a log collection policy in basic edit mode or advanced edit mode. You can use the basic edit mode to configure a simple log collection policy. If the basic edit mode does not meet your business requirements, you can enable the advanced edit mode. In advanced edit mode, you can flexibly configure a complex log collection policy.

   > **Note**
   > - You can configure multiple policies based on your business requirements.
   > - In advanced edit mode, you can edit policy statements. After you edit a policy statement, you cannot directly return to the basic edit mode.
   > - To return to the basic edit mode, you must delete all policy statements and save the changes. Then, click Collection Policy.

   - Configure a log collection policy in basic edit mode.

     a. In the **Add Policy** section, set the parameters and click **Add Policy**. The following table describes the parameters.

     > **Note** If you turn on **Default Collection Policy**, the last line of the collection policy is `accept "*"(Default Policy - Accept)`. If you turn off **Default Collection Policy**, the last line of the collection policy is `drop "*"(Default Policy - Discard)`.



Parameter	Description
Action	The action that is performed when Log Service collects logs based on the log collection policy. For more information, see Policy syntax.
Properties	The property of the log source. The available properties vary based on the log source that you use. For more information, see Supported Alibaba Cloud services.
Operator	The match mode that corresponds to an operator. If you select **Exact Match**, the operator is ==. For more information, see Policy syntax.
Property value	The value of the property. You can specify multiple values for a property.

b. In the **Added Policies** section, confirm the details of the log collection policy that you configured.

You can modify the policy and change the order of the policy.

- To modify the policy, click **Edit** on the right side of the policy.
- To change the order of the policy, click the upward or downward arrow on the right side of the policy.

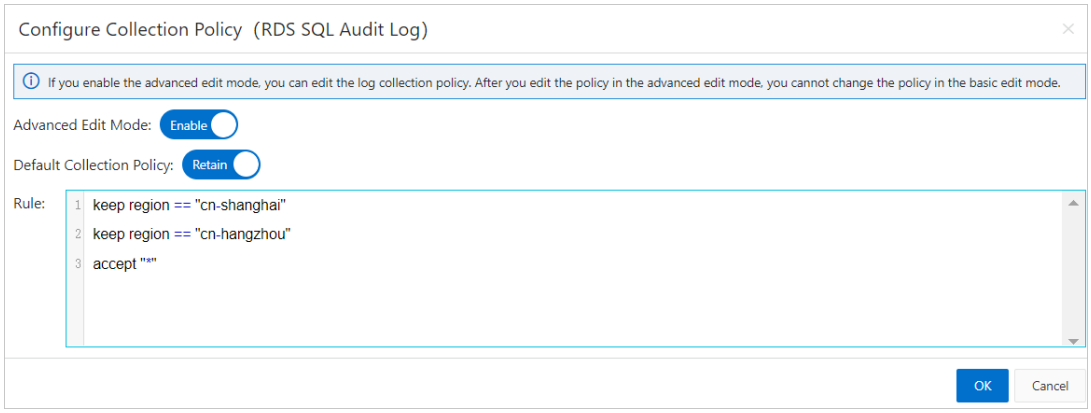
c. Confirm the settings and click **OK**.

- Configure a log collection policy in advanced edit mode.

a. Turn on **Advanced Edit Mode**.

b. In the **Rule** field, configure a log collection policy and click **OK**.

For information about the policy syntax, see Policy syntax.



6. On the **Global Configurations** page, click **OK**.

Policy syntax

- Actions
  - Keep: If the log source matches a policy, Log Service attempts to match the log source against the next policy and determines whether to collect logs based on subsequent policies. If the log source does not match the policy, Log Service does not collect logs and no longer attempts to match the log source against subsequent policies.
  - Drop: If the log source matches a policy, Log Service does not collect logs and no longer attempts to match the log source against subsequent policies. If the log source does not match the policy, Log Service attempts to match the log source against the next policy and determines whether to collect logs based on subsequent policies.
  - Accept: If the log source matches a policy, Log Service collects logs and no longer attempts to match the log source against subsequent policies. If the log source does not match the policy, Log Service attempts to match the log source against the next policy and determines whether to collect logs based on subsequent policies.
- Matching modes

Matching mode	Description
Exact match	Exact match is performed based on strings. ◦ Operator: ==. ◦ Example: **keep instance.db_type == "mysql"**. This policy evaluates to true for an ApsaraDB RDS for MySQL instance.
Wildcard match	Data is matched based on wildcard characters. The wildcard characters include asterisks (*) and question marks (?). An asterisk (*) specifies zero or multiple characters. A question mark (?) specifies one character. ◦ Operator: ==. ◦ Examples: ■ **keep instance.name == "backend*"**. This policy evaluates to true for an instance whose name starts with backend. ■ **keep instance.name == "active?"**. This policy evaluates to true for an instance whose name starts with active and a random character.
Regex match	Data is matched based on regular expressions. ◦ Operator: ~=. ◦ Example: **keep instance.name ~= "^\d+$"**. This policy evaluates to true for an instance whose name contains only digits. **Note** By default, Log Service performs partial match. To enable exact match, you must prefix a regular expression with a caret (^) and suffix the regular expression with a dollar sign ($).

Matching mode	Description
Numeric value comparison	The comparison of numeric values. ○ Operators: ■ Operators for direct comparison: greater-than (>), greater-than-or-equal to (>=), equal-to (=), less-than-or-equal-to (<=), and less-than (<). ■ Operators used to compare numeric values within a closed interval. Example: : [*, 100]. You can use an asterisk (*) to specify an infinite interval. ○ Examples: ■ **keep tag.level >= 2**. This policy evaluates to true for an instance whose value of the tag.level property is greater than or equal to 2. ■ **keep tag.level : [*, 10]**. This policy evaluates to true for an instance whose value of the tag.level property is less than or equal to 10. ■ **keep tag.level : [1, 10]**. This policy evaluates to true for an instance whose value of the tag.level property is within the closed interval [1, 10].
Logical operator	○ Keywords: ■ and, AND, and &&: The keywords are not case-sensitive. ■ or and OR: The keywords are not case-sensitive. ■ not, NOT, and exclamation point (!): The keywords are not case-sensitive. ○ Examples: ■ **keep (tag.level > 10) and (region == "cn-shanghai")**. This policy evaluates to true for an instance whose value of the tag.level property is greater than 10 and that resides in the China (Shanghai) region. ■ **keep (tag.level > 10) or (region == "cn-shanghai")**. This policy evaluates to true for an instance whose value of the tag.level property is greater than 10 or that resides in the China (Shanghai) region. ■ **keep not region == "cn-shanghai"**. This policy evaluates to true for an instance that does not reside in the China (Shanghai) region.

Matching mode	Description
Global match	If no property is specified in a log collection policy, the system matches log sources against all available properties for the policy. Examples: ○ **keep "abc"**. This policy evaluates to true for logs that contain the **abc** string. ○ **accept "*"**. This policy evaluates to true for all log sources. **Note** ○ If you use global match, you must enclose specified characters in double quotation marks (""). ○ Global match is available only in advanced edit mode.

- Character escape

  If a log collection policy contains special characters such as asterisks (*) and backslashes (\), you must escape the special characters. Example: **keep instance.name == "abc*"**. This policy evaluates to true for an instance whose name is abc*.

Common scenarios

- Collect the logs of instances that reside in specific regions

  In this example, only the logs of instances that reside in regions within the Chinese mainland are collected based on the configured collection policies.

```
# only scan cn region
keep region == "cn-*"
# accept by default
accept "*"
```

- Collect the logs of instances that have specified tags

  In this example, only the logs of instances whose value of the type tag is production are collected based on the configured collection policies. The value production is not case-sensitive.

```
# only scan "production" instances
keep tag.type ~= "(?i)^production$"
# accept by default
accept "*"
```

- Complex scenarios

  If the level: high tag is used in log collection policies, the logs of ApsaraDB RDS for MySQL instances, ApsaraDB RDS for SQL Server instances, and ApsaraDB RDS for PostgreSQL instances are collected. If the level: high tag is not used, only the logs of ApsaraDB RDS for MySQL instances are collected. The following code shows the log collection policies that are involved:

```
# accept all high level instances
accept tag.level == "high"
# only scan mysql
keep instance.db_type == "mysql"
# accept by default
accept "*"
```

1.6. Perform audit operations

This topic describes the audit operations that you can perform in the Log Audit Service application after logs are collected.

Prerequisites

- The Log Audit Service application is configured. For more information, see Enable log collection.
- Your account is granted the required permissions. For more information about how to grant permissions, see Configure the permission assistant feature.
  - To query logs or view reports, you must grant read permissions on the Log Audit Service application and the resources of related projects to your account.
  - To create reports, configure alerts, or make secondary access configurations, you must grant read and write permissions on the Log Audit Service application and the resources of related projects to your account.

View audit reports

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left navigation sidebar, click **Audit Report**.
4. Click the report that you want to view and go to the audit center.

   On the page that appears, you can view the reports. For information about how to manage a dashboard, see Overview.

   > **Note** For Object Storage Service (OSS), Server Load Balancer (SLB), PolarDB-X 1.0 and Virtual Private Cloud (VPC), if you do not turn on **Synchronization to Central Project** on the **Global Configurations** page, you can view the reports for different regions only on the **Regional** tab. If you turn on **Synchronization to Central Project**, you can view the reports also on the **Central** tab.

Query audit logs

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left navigation sidebar, click **Audit Query**.
4. Click the service whose audit logs you want to query and go to the query and analysis page.

   For more information about how to query and analyze data, see Query and analysis.

> **Note** For OSS, SLB, PolarDB-X 1.0 and VPC, if you do not turn on **Synchronization to Central Project** on the **Global Configurations** page, you can view the logs for different regions only on the **Regional** tab. If you turn on **Synchronization to Central Project**, you can view the logs also on the **Central** tab.

Manage Logstores

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. Choose **Audit Configurations > Access to Cloud Products > Global Configurations**.
4. Click the name of the project and go to the Logstores page.

What's next

After you complete log audit, you can ship data to third-party systems or use the systems to consume data. The third party systems refer to systems except for Log Service.

- Data shipping

  You can ship data to third-party systems. The systems include OSS, MaxCompute, AnalyticDB for MySQL, Time Series Database (TSDB), Splunk, and security information and event management (SIEM) tools. For more information, see Data shipping.

- Data consumption

  You can consume log data in real time by using third-party stream processing systems. The systems include Storm, Flume, Application Real-Time Monitoring Service (ARMS), Blink, Logstash, Spark Streaming, CloudMonitor, and consumer groups. For more information, see Real-time consumption.

1.7. View global data

This topic describes how to view the global data of monitored Alibaba Cloud services in the Log Audit Service application.

View the global data of log audit

1. Log on to the Log Service console.
2. In the **Log Application** section, click **Log Audit Service**.
3. Choose **Audit Configurations > Access to Cloud Products > Overall Dashboard**. On the page that appears, you can view the global data for log audit.

View the global data of monitored cloud services

1. Log on to the Log Service console.
2. In the **Log Application** section, click **Log Audit Service**.
3. Choose **Audit Report > Central**. Click the Display icon of the target service. From the menu of the target service, select Overall View.

> **Note** The Log Audit Service application supports the global data of Object Storage Service (OSS), Server Load Balancer (SLB), and ApsaraDB for RDS.

Report details

- Log Audit Overall Data View

Chart	Description	Remarks
Active Accounts	The total number of monitored accounts for log audit.	None
Total Logs, Logs For An Hour, and Logs For One Day	The log statistics in different periods.	The statistics may be delayed for no more than half an hour.
Overall Data View and Product Logs Distribution	The overview of collected logs of all services for which log audit is enabled.	The statistics may be delayed for no more than half an hour.
Overall Logs Trend and Product Logs Trend	The variation of the total amount of collected logs and the variation of collected log statistics of each service in the past 30 days.	The log statistics of the current day may be delayed for 1 hour.

- OSS Overall Data View

Chart	Description
Total Logs, Logs For An Hour, and Logs For One Day	The log statistics in different periods.
Total Access Logs, Access Logs For An Hour, and Access Logs For One Day	The access log statistics in different periods.
Total Metering Logs, Metering Logs For An Hour, and Metering Logs For One Day	The metering log statistics in different periods.
OSS Overall Info	The overall information about OSS buckets for which log audit is enabled.
Overall Logs Trend and Product Logs Trend	The variation in the total amount of collected logs and the variation in the amount of collected logs of each type in the past 30 days.

- SLB Overall Data View

Chart	Description
Total Logs, Logs For An Hour, and Logs For One Day	The log statistics in different periods.
Classic Network Total Logs, Classic Network Logs For An Hour, and Classic Network Logs For One Day	The log statistics of SLB instances that reside in the classic network in different periods.
VPC Network Total Logs, VPC Network Logs For An Hour, and VPC Network Logs For One Day	The log statistics of SLB instances that reside in virtual private clouds (VPCs) in different periods.

Chart	Description
SLB Overall Info	The overall information about SLB instances for which log audit is enabled.
Overall Logs Trend and Product Logs Trend	The variation of the total amount of logs and the variation of log statistics in each type of network in the past 30 days.

- RDS Overall Data View

Chart	Description
Total Logs, Logs For An Hour, and Logs For One Day	The log statistics in different periods.
MySQL Total Logs, MySQL Logs For An Hour, and MySQL Logs For One Day	The log statistics of ApsaraDB RDS for MySQL in different periods.
PgSQL Total Logs, PgSQL Logs For An Hour, and PgSQL Logs For One Day	The log statistics of ApsaraDB RDS for PostgreSQL in different periods.
MSSQL Total Logs, MSSQL Logs For An Hour, and MSSQL Logs For One Day	The log statistics of ApsaraDB RDS for SQL Server in different periods.
RDS Overall Info	The overall information about ApsaraDB for RDS instances for which log audit is enabled.
Overall Logs Trend and Product Logs Trend	The variation of the total amount of logs and the variation of log statistics of each type of ApsaraDB for RDS instance in the past 30 days.

1.8. Alerting

1.8.1. Configure alerts

The Log Audit Service application provides built-in alert rules. You can enable the alert instances of alert rules to monitor logs in real time. This topic describes how to configure alerts.

Prerequisites

The audit feature is enabled on the **Global Configurations** page for related cloud services. For more information, see Enable log collection.

Context

The Log Audit Service application provides built-in resources such as alert rules, alert policy, action policy, user group, and alert templates. You can use these built-in resources based on the following rules:

- You can specify the built-in alert policy in an alert rule.

> **Note** The built-in alert rules that are provided by the Log Audit Service application are associated with the built-in alert policy. You cannot disassociate the built-in alert policy from the alert rules or associate other alert policies with the alert rules

- You can specify the built-in action policy in the built-in action policy.
- You can specify the built-in user group and specify a built-in alert template in the built-in action policy.

Configuration process

You can use built-in resources or custom resources to configure alerts. The following process shows how to configure alerts.

- Use built-in resources

  To configure alerts in an efficient manner and to receive alert notifications by using voice calls, SMS messages, or emails, perform the following operations:

  i. Create users
  ii. Add users to the built-in user group
  iii. Enable alert instances

- Use custom resources

  To create custom resources and use the custom resources to configure alerts based on your business requirements, perform the following operations:

  i. Create users and user groups
  ii. Create an alert template
  iii. Create an action policy
  iv. Modify the action policy that is associated with the built-in alert policy
  v. Configure whitelists
  vi. Enable alert instances

The built-in resources that are provided by Log Service can be applied to most alerting scenarios. You can use built-in resources or custom resources based on your business requirements. In this example, built-in resources are used to configure alerts.

Step 1: Create users

1.
2. In the **Log Application** section, click **Log Audit Service**.
3. In the left-side navigation pane, choose **Audit Alert > User Management > User**.
4. Create users.

   For more information, see Create users and user groups.

Step 2: Add users to the built-in user group

1. In the left-side navigation pane, choose **Audit Alert > User Management > User Group**.
2. In the User Groups list, find the built-in user group whose ID is **sls.app.audit.builtin** and click **Edit**

in the Actions column.

3. In the **Edit User Group** dialog box, add the users that you created from the **Available Members** section to the **Selected Members** section. Then, click **OK**.

Step 3: Enable alert instances

1. In the left-side navigation pane, choose **Audit Alert > Policy Settings > Alert Rules**.
2. In the Alert Rules list, find the alert rule that you want to use and click **Enable** in the Actions column.

   After you enable an alert instance, Log Service monitors the Log Audit Service application in real time. To enable multiple alert instances, click **Add**.

   For more information about built-in alert rules, see Overview.

Related operations

Operation	Description
Configure whitelists	You can configure whitelists for specific alert rules. This way, alerts are not triggered by specific users, instance IDs, or IP addresses. The whitelist configurations vary based on alert rules. For more information, see Overview.
Disable alert instances	If you disable an alert instance, the status in the **Status** column of the alert instance changes to **Not Enabled**, and no more alerts are triggered based on the alert instance. The configurations of the alert instance are not deleted. If you want to re-enable the alert instance to monitor data, you do not need to reconfigure the parameters of the alert instance.
Pause alert instances	If you pause an alert instance, no alerts are triggered within a specified period of time based on the alert instance.
Resume alert instances	You can resume the alert instances that are paused.
Delete alert instances	If you delete an alert instance, the status in the **Status** column of the alert instance changes to **Not Created**. The configurations of the alert instance such as the settings of an Alibaba Cloud account are deleted. If you want to re-enable the alert instance to monitor data, you must set the parameters of the alert instance again.
Upgrade alert instances	If a major upgrade is released for alert rules or if additional configurations are required after alert rules are upgraded, you are prompted to upgrade alert rules. In most cases, Log Service automatically upgrades alert rules.
Initialize alerts	If the assets generated during alert initialization are deleted by mistake or if the alert assets fail to be initialized for the first time, you can perform this operation to forcibly re-initialize the alert assets.

Operation	Description
Modify the action policy that is associated with the built-in alert policy	If you want to use a custom action policy, you must create the custom action policy, and then modify the action policy that is associated with the built-in alert policy on the **Alert Policy** page.

1.8.2. Alert rule

1.8.2.1. Overview

This topic describes the built-in alert rules of the Log Audit Service application. You can use the alert rules to monitor the operation compliance, account security, permissions, and traffic security of the Log Audit Service application. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

Type	Alert rule
Log audit compliance	Cloud Security Center Log Audit Configuration Check
	RDS Log Audit Configuration Check
	PolarDB(DRDS) Log Audit Configuration Check
	K8s Log Audit Configuration Check
	Web Application Firewall (WAF) Log Audit Configuration Check
	Bastion Log Audit Configuration Check
	APIGateway Log Audit Configuration Check
	Cloudfirewall Log Audit Configuration Check
	Log Audit Status Check
	ActionTrail Log Audit Configuration Check
Account security	RAM Sub-Account Login without MFA Alert
	RAM Password Expiration Policy Exception Alert
	Root Account Login without MFA Alert
	RAM Password Login Retry Policy Exception Alert
	Root Account Frequent Login Alert

Account security Type	Alert rule
	RAM History Password Check Policy Exception Alert
	KMS Key Configuration Change Alert
	Account Continuous Login Failure Alert
	Root Account AK Usage Detection
	RAM Password Length Policy Exception Alert
Permission control	OSS Bucket Authority Change Alert
	RAM Policy Change Alert
	RAM Policy Abnormal Attach Alert
OSS operation compliance	OSS Bucket Encryption Shutdown Alert
	OSS Newly Created Bucket Encryption Not Enabled Alert
	OSS Bucket Logging Shutdown Alert
	OSS Newly Created Bucket Logging Not Enabled Alert
Operation compliance of RDS instances	RDS Instance SQL Insight Disabled Alert
	RDS Instance Access Whitelist Abnormal Setting Alert
	Newly Created RDS Instance's SSL Not Enabled AlertNot CreatedEnable Settings
	Newly Created RDS Instance's TDE Not Enabled Alert
	RDS Instance SSL Disabled Alert
	RDS Instance Configuration Change Alert
Server Load Balancer (SLB) operation compliance	SLB Modification Protection Shutdown Alert
	SLB Health Check Shutdown Alert
Operation compliance of ECS instances	ECS Disk Encryption Shutdown Alert
	ECS Automatic Snapshot Strategy Shutdown Alert
	Security Group Configuration Change Alert
	ECS Network Type Check
Operation compliance of VPCs	VPC Network Routing Change Alert
	VPC Flow Log Abnormally Configured Alert

Type	Alert rule
	VPC Configuration Change Alert
Operation compliance of Cloud Firewall	Cloudfirewall Control Policy Change Alert
API calls	Unauthorized Api Call Alert
Operation compliance of TDI	TDI Webpage Anti-tampering Disabled Alert
Kubernetes security	Too Many K8s Warning Events Alert
	K8s Frequent Delete Event Alert
	Too Many K8s Error Events Alert
Security of RDS instances	RDS Slow SQL detection
	RDS Data Mass Deletion Alert
	Detection of RDS Visit through Internet
	RDS Query SQL Average Execution Time Monitoring
	RDS Instance Update Peak Monitoring
	RDS Instance Query Peak Monitoring
	RDS Instance Released Alert
	RDS Frequent Visit IP Detection
	RDS Update SQL Average Execution Time Monitoring
	Too Many RDS Login Failures Alert
	Rds Mass Data Update Event Alert
	RDS Dangerous SQL Execution Alert
	Too Many RDS SQL Execution Errors Alert
Flow security of SLB	Inspection of SLB Abnormal Response Length
	Inspection of SLB Abnormal Request Length
	SLB Average Response Delay Too High Alert
	SLB HTTP Access Protocol Enabled Alert
	Load Balance Access UV Anomaly Inspection
	Load Balance Access PV Anomaly Inspection

Type	Alert rule
Flow security of API Gateway	APIgateway Server Average Delay Too High Alert
	APIGateway Backend Server Error Rate Too High Alert
	APIgateway Request Success Rate Too Low Alert
Security of OSS traffic	OSS Inflow Anomaly Inspection
	OSS Bucket Valid Request Rate Too Low Alert
	Detection of OSS Bucket Visit through Internet
	OSS Access PV Anomaly Inspection
	OSS Flow Anomaly Inspection
	OSS Outflow Anomaly Inspection
	OSS Access UV Anomaly Inspection
The security of Kubernetes traffic	Too Many K8s Illegal Access Alert
	K8s Ingress Average Request Latency Too High Alert
	K8s Ingress Response Delay Too High Alert
	K8s Ingress Request Success Rate Too Low Alert
Security of OSS data	OSS Bucket Account Access Control
	OSS Object Frequent Deletion Alert
Data security of NAS	NAS Error Operation Detection
	NAS Mass Deletion Alert
Security events of WAF	Application Firewall Valid Request Rate Too Low Alert
	Too Many Attacks on Hosts Protected by WAF Alert
TDI security events	Too Many High-Priority Alarms In Cloud Security Center
	Too Many New Vulnerabilities In Cloud Security Centers
	Cloud Security Center Valid Request Rate Too Low Alert
	Too Many New Alarms In Cloud Security Center
	Cloud Security Center Request Success Rate Too Low
	Cloudfirewall Outflow Block Alert

Security events of Cloud Firewall Type	Alert rule
	Cloudfirewall Inflow Block Alarm

1.8.2.2. Log audit compliance

This topic describes the alert rules for the log audit compliance of multiple Alibaba Cloud services. These services include Object Storage Service (OSS), ApsaraDB RDS, PolarDB, Server Load Balancer (SLB), Apsara File Storage NAS (NAS), and Container Service for Kubernetes. You can configure and enable alert rules in the Log Service console. This allows you to monitor the log audit compliance of these services. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- Cloud Security Center Log Audit Configuration Check
- RDS Log Audit Configuration Check
- Log Audit Status Check
- PolarDB(DRDS) Log Audit Configuration Check
- K8s Log Audit Configuration Check
- ActionTrail Log Audit Configuration Check
- OSS Log Audit Configuration Check
- Web Application Firewall (WAF) Log Audit Configuration Check
- Bastion Log Audit Configuration Check
- NAS (File Storage) Log Audit Configuration Check
- APIGateway Log Audit Configuration Check
- SLB Log Audit Configuration Check
- Cloudfirewall Log Audit Configuration Check

Cloud Security Center Log Audit Configuration Check

ID	sls_app_audit_cis_at_sas_audit_check
Name	Cloud Security Center Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for Security Center logs. If the audit switch is turned off for Security Center logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	Min storage duration(ttl): The minimum duration for which Security Center logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. On the page that appears, turn on the Audit Logs switch next to Security Center(SAS). Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

RDS Log Audit Configuration Check

ID	sls_app_audit_cis_at_rds_audit_check
Name	RDS Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for RDS logs. If the audit switch is turned off for the RDS logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which RDS logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. On the page that appears, turn on the SQL Audit Log switch next to RDS. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

Log Audit Status Check

ID	sls_app_audit_cis_at_audit_status_check
Name	Log Audit Status Check
Version	1

Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks the status of the log audit service. If the status is abnormal, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	None
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Status Dashboard**. On the page that appears, check the status of the log audit service and identify the cause of the abnormal status.
Prerequisites	None

PolarDB(DRDS) Log Audit Configuration Check

ID	sls_app_audit_cis_at_drds_audit_check
Name	PolarDB(DRDS) Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for PolarDB logs. If the audit switch is turned off for the PolarDB (DRDS) logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which PolarDB logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. On the page that appears, turn on the Audit Log switch next to PolarDB. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

K8s Log Audit Configuration Check

ID	sls_app_audit_cis_at_k8s_audit_check
Name	K8s Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for Kubernetes logs, including Kubernetes audit logs, Kubernetes events, and Ingress access logs. If the audit switch is turned off for K8s logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which K8s logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. On the page that appears, turn on the Kubernetes Audit Log switch, K8s Event Center switch and Ingress Log switch next to Kubernetes. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

ActionTrail Log Audit Configuration Check

ID	sls_app_audit_cis_at_actiontrail_audit_check
Name	ActionTrail Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for ActionTrail logs. If the audit switch is turned off for the of Action Trail logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which Action Trail logs are stored. Default value: 180 days.

External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Operations Log switch next to ActionTrail. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

OSS Log Audit Configuration Check

ID	sls_app_audit_cis_at_oss_audit_check
Name	OSS Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for Object Storage Service (OSS) logs, including access logs and metering logs. If the audit switches are turned off for OSS logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which OSS logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Metering Log switch and the Access Log switch next to OSS. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

Web Application Firewall (WAF) Log Audit Configuration Check

ID	sls_app_audit_cis_at_waf_audit_check
Name	Web Application Firewall (WAF) Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance

Usage	Checks whether log audit is properly configured in the Log Audit Service application for the Web Application Firewall (WAF) logs. If the audit switch is turned off for Web Application Firewall (WAF) logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which WAF Logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Access Log switch next to Web Application Firewall (WAF). Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

Bastion Log Audit Configuration Check

ID	sls_app_audit_cis_at_bastion_audit_check
Name	Bastion Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for Bastionhost logs. If the audit switch is turned off for the Bastionhost log or its storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which Bastionhost log is stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Operations Log switch next to Bastion Host. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

NAS (File Storage) Log Audit Configuration Check

ID	sls_app_audit_cis_at_nas_audit_check
Name	NAS (File Storage) Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for the Apsara File Storage NAS logs. If the audit switch is turned off for NAS (file storage) logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which NAS (file storage) logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Access Log switch next to NAS. Make sure that the storage duration is greater than the value of the Min storage duration(ttl)parameter.
Prerequisites	None

APIGateway Log Audit Configuration Check

ID	sls_app_audit_cis_at_apigateway_audit_check
Name	APIGateway Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for the API Gateway logs. If the audit switch is turned off for API Gateway logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which API Gateway logs are stored. Default value: 180 days.

External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Access Log switch next to API Gateway. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

SLB Log Audit Configuration Check

ID	sls_app_audit_cis_at_slb_audit_check
Name	SLB Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance
Usage	Checks whether log audit is properly configured in the Log Audit Service application for SLB logs. If the audit switch is turned off for SLB logs or the storage duration is smaller than the value of the Min storage duration(ttl) parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which SLB logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Lay-7 Access Log switch next to SLB. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

Cloudfirewall Log Audit Configuration Check

ID	sls_app_audit_cis_at_cloudfirewall_audit_check
Name	Cloudfirewall Log Audit Configuration Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Log Audit Compliance

Usage	Checks whether log audit is properly configured in the Log Audit Service application for Cloud Firewall logs. If the audit switch is turned off for the Cloud Firewall log or its storage duration is smaller than the value of the Min storage duration(ttl)parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Min storage duration(ttl): The minimum duration for which Cloud Firewall logs are stored. Default value: 180 days.
External Configurations	None
Solution	On the Log Audit Service page, choose **Audit Configurations > Access to Cloud Products > Global Configurations**. Turn on the Internet Access Log switch next to Cloud Firewall. Make sure that the storage duration is greater than the value of the Min storage duration(ttl) parameter.
Prerequisites	None

1.8.2.3. Account security

This topic describes the alert rules for account security. You can configure and enable alerts in the Log Service console. This allows you to monitor account security issues. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- RAM Sub-Account Login without MFA Alert
- RAM Password Expiration Policy Exception Alert
- Root Account Login without MFA Alert
- RAM Password Login Retry Policy Exception Alert
- Root Account Frequent Login Alert
- RAM History Password Check Policy Exception Alert
- KMS Key Configuration Change Alert
- Account Continuous Login Failure Alert
- Root Account AK Usage Detection
- RAM Password Length Policy Exception Alert

RAM Sub-Account Login without MFA Alert

ID	sls_app_audit_cis_at_ram_mfa
Name	RAM Sub-Account Login without MFA Alert

Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether multi-factor authentication (MFA) is disabled for a Resource Access Management (RAM) user who log to the console of an Alibaba Cloud service. When a RAM user logs onto the console, MFA must be enabled for the RAM user. In addition, the number of logons without MFA must be less than or equal to the specified Max logins parameter. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Max logins: The maximum number of logons allowed every five minutes for a RAM user whose MFA is disabled. Default value: 0.
External Configurations	You can configure a whitelist of RAM users who can log on to the consoles of Alibaba Cloud services without the need to enable MFA. If MFA is disabled for a RAM user on the whitelist when the RAM user logs on to the console of an Alibaba Cloud Service, no alert is triggered.
Solution	Make sure that the number of logons without MFA for of a RAM user within 5 minutes is less than or equal to the specified Max logins parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM Password Expiration Policy Exception Alert

ID	sls_app_audit_cis_at_pwd_expire_policy
Name	RAM Password Expiration Policy Exception Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether the validity period specified in a RAM password policy is valid. In the RAM password policy, the validity period of a RAM password less than or equal to the specified Max validity period parameter in the alert rule. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 5 minutes.
Time Range	The data of the last 5 minutes is checked.

Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Max validity period. The default value is 90 days. To meet the Center for Internet Security (CIS) rules of Alibaba Cloud, we recommend that you set the value to 90 or less.
External Configurations	None
Solution	Make sure that the validity period in the RAM password policy is less than or equal to the specified Max validity period parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Root Account Login without MFA Alert

ID	sls_app_audit_cis_at_root_mfa
Name	Root Account Login without MFA Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors the logons of a root user to the console without MFA being enabled. When a root user wants to log on to the console, the MFA must be enabled. Also, the number of logons without MFA must be smaller than or equal to the specified Max logins parameter. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Max logins: The maximum number of logons allowed on a single day for a root user whose MFA is disabled. Default value: 0.
External Configurations	You can configure a whitelist of root users who are allowed to log on without MFA. The root users on the whitelist can log on for an unlimited number of times without MFA. No alert is triggered by such logons.
Solution	Make sure that the number of non-MFA logins of the root user within 5 minutes is smaller than or equal to the specified Max logins parameter.

Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM Password Login Retry Policy Exception Alert

ID	sls_app_audit_cis_at_pwd_login_attemp_policy
Name	RAM Password Login Retry Policy Exception Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether the logon retry policy specified in a RAM password policy is valid. In the RAM password policy, the number of failed logons attempts within one hour due to invalid passwords cannot be greater than the specified Max login failures/h parameter in the alert rule. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 5 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Max login failures/h: In the RAM password policy, the maximum number of failed logons due to invalid passwords that are allowed within a single hour. Default value: 5. To meet the Center for Internet Security (CIS) rules of Alibaba Cloud, we recommend that you set the value to 5.
External Configurations	None
Solution	You can reset the number of failed logons that are allowed within one hour due to invalid passwords in the RAM password policy. Make sure that it is smaller than or equal to the specified Max login failures/h parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Root Account Frequent Login Alert

ID	sls_app_audit_cis_at_root_login
Name	Root Account Frequent Login Alert
Version	1

Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors frequent logins of a root user. Root users cannot frequently log on to the console of an Alibaba Cloud service. If the number of logons of a root user within 5 minutes exceeds the Max login Times parameter, an alert is triggered.
Check Frequency	Fixed interval: 5 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Max login Times: The maximum number of logons that are allowed for a root user within 5 minutes. Default value: 2.
External Configurations	A whitelist of root users who are allowed to log on frequently. The root users on the whitelist can log on for an unlimited number of times within 5 minutes. No alert is triggered by such logons.
Solution	On a daily basis, you can limit the number of frequent logons of the root user. Make sure that it is smaller than or equal to the specified Max login Times parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM History Password Check Policy Exception Alert

ID	sls_app_audit_cis_at_pwd_reuse_policy
Name	RAM History Password Check Policy Exception Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether the historical password check policy specified in the RAM password policy is valid. In a historical password check policy, the previous N passwords cannot be reused. You can specify the minimum value of N in the parameter settings of the alert rule. If the value in the historical password policy is less than this threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Minimum password reuse value: The minimum value of N in the **previous N passwords is prohibited** parameter in the historical password check of the RAM password policy. Default value: 4. To meet the Center for Internet Security (CIS) rules of Alibaba Cloud, we recommend that you set the value to 4.
External Configurations	None
Solution	Make sure that the value of N in the **previous N passwords is prohibited** is greater than or equal to the specified Minimum password reuse value parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

KMS Key Configuration Change Alert

ID	sls_app_audit_cis_at_ak_conf_change
Name	KMS Key Configuration Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether the key configuration in Key Management Service (KMS) is changed. When the key configuration in KMS is changed (such as deleted or disabled), an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of RAM users who are allowed to modify the key configuration in KMS. RAM users on the whitelist can modify the key configuration in KMS without triggering an alert.
Solution	Prohibit the RAM users that are not included in the whitelist from modifying the key configuration.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Account Continuous Login Failure Alert

ID	sls_app_audit_cis_at_abnormal_login_count
Name	Account Continuous Login Failure Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors the number of consecutive logon failures within a specific period of time. When the number of failed logons within 5 minutes is greater than the specified Failed logins parameter, an alert is triggered.
Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Failed logins: The maximum number of failed logons that is allowed within 5 minutes for an account. Default value: 5.
External Configurations	None
Solution	Make sure that the number of failed logons within 5 minutes is less than or equal to the Failed logins parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Root Account AK Usage Detection

ID	sls_app_audit_cis_at_root_ak_usage
Name	Root Account AK Usage Detection
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors the usage of the AccessKey pair of a root account. Root users cannot create or use AccessKey pairs for their root accounts. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8

External Configurations	You can configure a whitelist of root users who are allowed to use AccessKey pairs. Root users on the whitelist can use AccessKey pairs without triggering an alert.
Solution	Make sure that the Root account AccessKey pair is not used.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM Password Length Policy Exception Alert

ID	sls_app_audit_cis_at_pwd_length_policy
Name	RAM Password Length Policy Exception Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Account Security
Usage	Monitors whether the minimum password length specified in the RAM password policy is valid. In the RAM password policy, the minimum length of a RAM password must be greater than or equal to the value of the specified Min password length parameter. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 5 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Min password length: The minimum value of the setting of the minimum password length in the password policy Default value: 14. To meet the Center for Internet Security (CIS) rules of Alibaba Cloud, we recommend that you set the value to 14.
External Configurations	None
Solution	You can reset the minimum password length in the RAM password policy. Make sure that it is greater than or equal to the specified Min password length parameter.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.4. Permission control

This topic describes the alert rules for permission control. These alert rules include the alert rules that you can use to monitor the changes of RAM policies, unexpected attachments of RAM policies, and changes of OSS bucket permissions. You can configure and enable alert rules in the Log Service console. This allows you to monitor permission control issues. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- OSS Bucket Authority Change Alert
- RAM Policy Change Alert
- RAM Policy Abnormal Attach Alert

OSS Bucket Authority Change Alert

ID	sls_app_audit_cis_at_oss_policy_change
Name	OSS Bucket Authority Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Permission Control
Usage	Monitors the change of OSS Bucket permission. Changes of OSS Bucket permission will trigger an alert.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of RAM users who can change the permissions of OSS buckets. If the RAM users on the whitelist change the permissions of OSS bucket, no alert is triggered.
Solution	Use only the RAM users who are included in the whitelist to change the permissions of OSS buckets.
Prerequisites	The **Operations Log** switch next to Action Trail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM Policy Change Alert

ID	sls_app_audit_cis_at_ram_policy_change
Name	RAM Policy Change Alert

Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Permission Control
Usage	Monitors the changes of RAM policy. If a RAM policy is changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6
External Configurations	You can configure a whitelist of RAM users who can change RAM policies. If the RAM users on the whitelist change RAM policies, no alert is triggered.
Solution	Disable the change of RAM policy for RAM users that are not included in the whitelist.
Prerequisites	The **Operations Log** switch next to Action Trail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

RAM Policy Abnormal Attach Alert

ID	sls_app_audit_cis_at_ram_policy_attach
Name	RAM Policy Abnormal Attach Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Permission Control
Usage	Monitors whether RAM policies are unexpectedly attached to RAM users. You can attach RAM policies only to RAM user groups or RAM roles. If you attach RAM policies to RAM users, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6
External Configurations	You can configure a whitelist of RAM users to whom RAM policies can be attached. RAM policies can be attached to RAM users on the whitelist without triggering an alert.
Solution	Attach RAM policies to user groups or roles instead of users.

Prerequisites	The **Operations Log** switch next to Action Trail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.5. OSS operation compliance

This topic describes alert rules for OSS operation compliance. The alert rules include OSS Bucket Encryption Shutdown and OSS Newly Created Bucket Encryption Not Enabled. You can configure and enable alert rules in the Log Service console. This allows you to monitor the issues of OSS operational compliance. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- OSS Bucket Encryption Shutdown Alert
- OSS Newly Created Bucket Encryption Not Enabled Alert
- OSS Bucket Logging Shutdown Alert
- OSS Newly Created Bucket Logging Not Enabled Alert

OSS Bucket Encryption Shutdown Alert

ID	sls_app_audit_cis_at_oss_encry_config
Name	OSS Bucket Encryption Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, OSS Operation Compliance
Usage	Monitors the encryption of OSS Bucket is disabled. All OSS buckets must be encrypted on the server side, and you are not recommended shutting down the encryption feature. If you disable the encryption, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	A whitelist of RAM users who can shut down the encryption for OSS buckets. RAM users on the whitelist can shut down encryption of OSS buckets without triggering an alert.
Solution	Enable the encryption of OSS buckets for accounts that are not included in the whitelist.

Prerequisites	The **Access Log** switch next to OSS is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

OSS Newly Created Bucket Encryption Not Enabled Alert

ID	sls_app_audit_cis_at_oss_bucket_encry_off
Name	OSS Newly Created Bucket Encryption Not Enabled Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, OSS Operation Compliance
Usage	Monitors whether the encryption of newly created OSS buckets is disabled. You must enable encryption when OSS Bucket is created or within one hour after it is created. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 1 hour is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of RAM users can keep the encryption of newly created OSS buckets disabled. RAM on the whitelist users can keep the encryption of newly created OSS buckets disabled.
Solution	You can turn on the encryption switch when the OSS bucket is created or enable it soon, within one hour after the creation.
Prerequisites	The **Access Log** switch next to OSS is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

OSS Bucket Logging Shutdown Alert

ID	sls_app_audit_cis_at_oss_log_config
Name	OSS Bucket Logging Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, OSS Operation Compliance
Usage	Monitors the access logs of OSS Bucket are disabled. You are recommended to enable all access logs of OSS Bucket. If you disable access logs of OSS bucket, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.

Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6
External Configurations	You can configure a whitelist of RAM users who can shut down access logs of OSS buckets. Users can shut down access logs of OSS buckets in RAM users on the whitelist without triggering an alert.
Solution	Enable the access logs of OSS buckets for RAM users that are not included in the whitelist.
Prerequisites	The **Access Log** switch next to OSS is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

OSS Newly Created Bucket Logging Not Enabled Alert

ID	sls_app_audit_cis_at_oss_log_off
Name	OSS Newly Created Bucket Logging Not Enabled Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, OSS Operation Compliance
Usage	Monitors whether the access logs of OSS Bucket are disabled. You must enable the access logs of OSS Bucket as soon as it is created. If you do not enable access logs for OSS Bucket within one hour after it is created, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 1 hour is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6
External Configurations	You can configure a whitelist of RAM users who can keep the access logs of newly created OSS buckets disabled. RAM users on the whitelist can keep the access logs of newly created OSS buckets.
Solution	Turn on the access log switch within one hour after the OSS bucket is created.
Prerequisites	The **Access Log** switch next to OSS is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.6. Operation compliance of RDS instances

This topic describes the alert rules for the operation compliance of RDS instances. You can configure and enable alert rules in the Log Service console to monitor the operation compliance of RDS instances. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- RDS Instance SQL Insight Disabled Alert
- RDS Instance Access Whitelist Abnormal Setting Alert
- Newly Created RDS Instance's SSL Not Enabled AlertNot CreatedEnable Settings
- Newly Created RDS Instance's TDE Not Enabled Alert
- RDS Instance SSL Disabled Alert
- RDS Instance Configuration Change Alert

RDS Instance SQL Insight Disabled Alert

ID	sls_app_audit_cis_at_rds_sql_audit
Name	RDS Instance SQL Insight Disabled Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitors whether the SQL Explorer feature is disabled for an RDS instance. The SQL Explorer feature must be enabled for RDS instances. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can disable the SQL Explorer feature for RDS instances. If the SQL Explorer feature is disabled by an account on the whitelist, no alert is triggered.
Solution	Do not disable the SQL Explorer feature for an RDS instance by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance Access Whitelist Abnormal Setting Alert

ID	sls_app_audit_cis_at_rds_access_whitelist

Name	RDS Instance Access Whitelist Abnormal Setting Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitors whether the whitelist of IP addresses to access RDS instances is invalid. The IP address on the whitelist to access an RDS instance cannot be set to 0.0.0.0. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts. If an RDS instance belongs to an account on the whitelist and the whitelist of IP addresses to access the instance is set to 0.0.0.0, no alert is triggered.
Solution	Allow only the RDS instance that belongs to an account on the whitelist to set the whitelist IP address to 0.0.0.0
Prerequisites	The **Operations Log** switch is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Newly Created RDS Instance's SSL Not Enabled AlertNot CreatedEnable Settings

ID	sls_app_audit_cis_at_rds_ssl_off
Name	Newly Created RDS Instance's SSL Not Enabled AlertNot CreatedEnable Settings
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitors whether the SSL feature is disabled for newly created RDS instances. We recommend that you enable the SSL feature within 1 hour after you create an RDS instance. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last hour is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.

External Configurations	You can specify a whitelist of accounts. If an RDS instance belongs to an account on the whitelist and the SSL feature is not enabled for the instance, no alert is triggered.
Solution	If an RDS instance does not belong to an account in the whitelist, we recommend that you enable the SSL feature within 1 hour after you create the instance.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Newly Created RDS Instance's TDE Not Enabled Alert

ID	sls_app_audit_cis_at_rds_tde_off
Name	Newly Created RDS Instance's TDE Not Enabled Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitor whether TDE is disabled for a newly created RDS instance. We recommend that you enable TDE within 1 hour after you create an RDS instance. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last hour is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6.
External Configurations	You can specify a whitelist of accounts. If an RDS instance belongs to an account on the whitelist and TDE is not enabled for the instance, no alert is triggered.
Solution	If an RDS instance does not belong to an account on the whitelist, we recommend that you enable TDE within 1 hour after you create the RDS instance.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance SSL Disabled Alert

ID	sls_app_audit_cis_at_rds_ssl_config
Name	RDS Instance SSL Disabled Alert

Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitors if the SSL feature is disabled for RDS instances. We recommend that you do not disable the SSL feature for RDS instances. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts. If an RDS instance belongs to an account on the whitelist and the SSL feature is disabled for the instance, no alert is triggered.
Solution	Do not disable the SSL feature for an RDS instance that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance Configuration Change Alert

ID	sls_app_audit_cis_at_rds_conf_change
Name	RDS Instance Configuration Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and RDS Operation Compliance
Usage	Monitors whether the configurations of RDS instances are changed. If the configurations of an RDS instance are changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Low-4.
External Configurations	You can specify a whitelist of accounts. If an RDS instance belongs to an account on the whitelist and the configurations of the instance are changed, no alert is triggered.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.

Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.7. Server Load Balancer (SLB) operation compliance

This topic describes the alert rules for the compliance of Server Load Balancer (SLB) operations. The alert rules include SLB health check shutdown and SLB modification protection shutdown. You can configure and enable alert rules in the Log Service console to monitor the compliance of SLB operations. If an alert is triggered, you can identify the compliance problems of SLB operations at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- SLB Modification Protection Shutdown Alert
- SLB Health Check Shutdown Alert

SLB Modification Protection Shutdown Alert

ID	sls_app_audit_cis_at_slb_mod_protec
Name	SLB Modification Protection Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, SLB Operation Compliance
Usage	Monitors whether the modification protection feature is disabled for Server Load Balancer (SLB) instances. The modification protection feature must be enabled for Server Load Balancer (SLB) instances. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of SLB instances whose modification protection feature can be disabled. If the modification protection feature is disabled for the SLB instances on the whitelist, no alert is triggered.
Solution	Enable the modification protection feature for the SLB instances that are not included in the whitelist.

Prerequisites	The **Access Log** switch next to API Gateway instance is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

SLB Health Check Shutdown Alert

ID	sls_app_audit_cis_at_slb_health_check
Name	SLB Health Check Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, SLB Operation Compliance
Usage	Monitors whether the health check feature is disabled for Server Load Balancer (SLB) instances. The health check feature must be enabled for Server Load Balancer instances. Otherwise, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of SLB instances whose health check feature can be disabled. If the health check feature is disabled for the SLB instances on the whitelist, no alert is triggered.
Solution	Enable the health check feature for the SLB instances that are not included in the whitelist.
Prerequisites	The **Access Log** switch next to API Gateway instance is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.8. Operation compliance of ECS instances

This topic describes the alert rules for the operation compliance of ECS instances. The alert rules are applicable to monitor the encryption status of ECS disks, the automatic snapshot policies of ECS instances, and the configurations of ECS security groups. You can configure and enable alert rules in the Log Service console to monitor the operation compliance of ECS instances. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- ECS Disk Encryption Shutdown Alert
- ECS Automatic Snapshot Strategy Shutdown Alert

- Security Group Configuration Change Alert
- ECS Network Type Check

ECS Disk Encryption Shutdown Alert

ID	sls_app_audit_cis_at_ecs_disk_encry_detection
Name	ECS Disk Encryption Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and ECS Operation Compliance
Usage	Monitors the encryption status of ECS disks. ECS disks are encrypted on the server side. If the encryption is disabled, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can disable the encryption feature of an ECS disk. If the encryption feature of an ECS disk is disabled by an account on the whitelist, no alert is triggered.
Solution	Do not disable the encryption feature of an ECS disk by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

ECS Automatic Snapshot Strategy Shutdown Alert

ID	sls_app_audit_cis_at_ecs_auto_snapshot_policy
Name	ECS Automatic Snapshot Strategy Shutdown Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and ECS Operation Compliance
Usage	Monitors if the automatic snapshot policies of ECS instances are disabled. To back up data for a disk, we recommend that you use automatic snapshot policies. If the automatic snapshot policies of ECS instances are disabled, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can disable the automatic snapshot policy of a disk. If the automatic snapshot policy is disabled by an account on the whitelist, no alert is triggered.
Solution	Do not disable the automatic snapshot policy of a disk by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Security Group Configuration Change Alert

ID	sls_app_audit_cis_at_securitygroup_change
Name	Security Group Configuration Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and ECS Operation Compliance
Usage	Monitors if the configurations of ECS security groups are changed. If the configurations of ECS security groups are changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can change the configurations of ECS security groups. If the configurations of security groups are changed by an account on the whitelist, no alert is triggered.
Solution	Do not change the configurations of security groups by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

ECS Network Type Check

ID	sls_app_audit_cis_at_ecs_network_type

Name	ECS Network Type Check
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and ECS Operation Compliance
Usage	Monitors the network type of ECS instances We recommend that you create ECS instances over a virtual private cloud (VPC). If you create an ECS instance over a classic network, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6.
External Configurations	You can specify a whitelist of accounts that can create an ECS instance over a classic network. If an ECS instance is created over a classic network by an account on the whitelist, no alert is triggered.
Solution	Do not create an ECS instance over a classic network by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.9. Operation compliance of VPCs

This topic describes the alert rules for the operation compliance of a virtual private cloud (VPC). You can configure and enable alert rules in the Log Service console to monitor the operation compliance of VPCs. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- VPC Network Routing Change Alert
- VPC Flow Log Abnormally Configured Alert
- VPC Configuration Change Alert

VPC Network Routing Change Alert

ID	sls_app_audit_cis_at_vpc_route_change
Name	VPC Network Routing Change Alert
Version	1

Type	Cloud Platform, Alicloud, CIS Standard, and VPC Operation Compliance
Usage	Monitors the routing configurations of a VPC. If the configurations are changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Low-4.
External Configurations	You can specify a whitelist of accounts that can change the configurations of a VPC. If the configurations of a VPC are changed by an account on the whitelist, no alert is triggered.
Solution	Do not change the configurations of a VPC by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

VPC Flow Log Abnormally Configured Alert

ID	sls_app_audit_cis_at_vpc_flowlog_detection
Name	VPC Flow Log Abnormally Configured Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and VPC Operation Compliance
Usage	Monitors VPC flow logs. We recommend that you enable the flow log feature of a VPC. If the flow log feature is disabled or deleted, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can disable the flow log feature of a VPC. If the flow log feature of a VPC is disabled by an account on the whitelist, no alert is triggered.
Solution	Do not disable or delete the flow log feature of a VPC by using an account that is not included in the whitelist.

Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

VPC Configuration Change Alert

ID	sls_app_audit_cis_at_vpc_conf_change
Name	VPC Configuration Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and VPC Operation Compliance
Usage	Monitors whether the configurations of a VPC are changed. If the configurations of a VPC are changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Low-4.
External Configurations	You can specify a whitelist of accounts that can change the configurations of a VPC. If the configurations of a VPC are changed by an account on the whitelist, no alert is triggered.
Solution	Do not change the configurations of a VPC by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.10. Operation compliance of TDI

This topic describes the alert rules for the operation compliance of TDI. You can configure and enable alert rules in the Log Service console to monitor the operation compliance of TDI. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

TDI Webpage Anti-tampering Disabled Alert

ID	sls_app_audit_cis_at_unauth_apicall
Name	TDI Webpage Anti-tampering Disabled Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, and TDI Operation Compliance

Usage	Monitors whether the web tamper protection feature of Security Center is disabled. If the web tamper protection feature of Security Center is disabled, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist of accounts that can disable the web tamper protection feature of Security Center. If the web tamper protection feature of Security Center is disabled by an account on the whitelist, no alert is triggered.
Solution	Do not disable the web tamper protection feature of Security Center by using an account that is not included in the whitelist.
Prerequisites	The **Operations Log** switch of ActionTrail is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.11. Operation compliance of Cloud Firewall

This topic describes the alert rules for the operation compliance of Cloud Firewall. You can configure and enable alert rules in the Log Service console. This allows you to monitor the operation compliance of Cloud Firewall. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Cloudfirewall Control Policy Change Alert the operation compliance of Cloud Firewall

ID	sls_app_audit_cis_at_cloudfirewall_conf_change
Name	Cloudfirewall Control Policy Change Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, Cloudfirewall Operation Compliance
Usage	Monitors the control policy changes of Cloud Firewall. If a control policy of Cloud Firewall is changed, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6

External Configurations	You can configure a whitelist of accounts whose control policies of Cloud Firewall can be changed. If the control policy of Cloud Firewall for accounts on the whitelist is changed, no alert is triggered.
Solution	Disable the control policy change of Cloud Firewall for accounts that are not included in the whitelist.
Prerequisites	The **Operations Log** switch next to Action Trail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.12. API calls

This topic describes the alert rules for the API call. You can configure and enable alert rules in the Log Service console. This allows you to monitor API calls. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Unauthorized Api Call Alert

ID	sls_app_audit_cis_at_unauth_apicall
Name	Unauthorized Api Call Alert
Version	1
Type	Cloud Platform, Alicloud, CIS Standard, API Call
Usage	Monitors the number of unauthorized API calls. If the number of unauthorized API calls is greater than the specified Maximum times of unauthorized api call parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6 • Maximum times of unauthorized api call: The maximum number of unauthorized API calls that can be initiated from one IP address to one single service every two minutes. Default value: 5.
External Configurations	You can configure a whitelist of IP addresses from which unauthorized API calls can be initiated. Unauthorized API calls from the IP addresses on the whitelist do not trigger an alert.
Solution	Disable the initiation of a large number of unauthorized API calls from IP addresses that are not included in the whitelist.

Prerequisites	The **Operations Log** switch next to Action Trail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.13. Kubernetes security

This topic describes the alert rules for Kubernetes security, including excessive number of Kubernetes events and error messages and frequent delete events. You can configure and enable alerts in the Log Service console. This allows you to monitor Kubernetes security issues. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- Too Many K8s Warning Events Alert
- K8s Frequent Delete Event Alert
- Too Many K8s Error Events Alert

Too Many K8s Warning Events Alert

ID	sls_app_audit_container_at_k8s_warn
Name	Too Many K8s Warning Events Alert
Version	1
Type	Cloud Platform, Alicloud, Container Security, K8s Security
Usage	Monitors the number of warning events on a Kubernetes cluster. If the number of warning events on a Kubernetes cluster is greater than or equal to the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alarm Name: The name of the alert. The default value is Too Many K8s Warning Events Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: Medium-6
	• Threshold: The maximum number of warning events that are broadcast by a Kubernetes cluster every 2 minutes. Default value: 10.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• K8s Cluster Name: The name of the Kubernetes cluster that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the Kubernetes clusters that are configured in the Log Audit Service application.

External Configurations	None
Solution	You can check whether exceptions have occurred on clusters that broadcast a great number of warning events.
Prerequisites	The **K8s Event Center** switch next to Kubernetes is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

K8s Frequent Delete Event Alert

ID	sls_app_audit_container_at_k8s_del
Name	K8s Frequent Delete Event Alert
Version	1
Type	Cloud Platform, Alicloud Container, Security K8s, Security
Usage	Monitors frequent delete events on Kubernetes clusters. If a delete event on a Kubernetes cluster is greater than or equal to the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. The default value is K8s Frequent Delete Event Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum number of delete events that are allowed on a Kubernetes cluster every 2 minutes. Default value: 5.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• K8s Cluster Name: The name of the K8s cluster that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates all Kubernetes clusters within an Alibaba Cloud account.

External Configurations	None
Solution	Check whether exceptions have occurred on the Kubernetes cluster where delete events occur too frequently.
Prerequisites	The **K8s Event Center** switch next to Kubernetes is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Too Many K8s Error Events Alert

ID	sls_app_audit_container_at_k8s_err
Name	Too Many K8s Error Events Alert
Version	1
Type	Cloud Platform, Alicloud, Container Security, K8s Security
Usage	Monitors the error events of a Kubernetes cluster. If the number of error events on a Kubernetes cluster is greater than the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alarm Name: The name of the alert. The default value is Too Many K8s Error Events Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum number of error events that are broadcast by a Kubernetes cluster every 2 minutes. Default value: 5.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• K8s Cluster Name: The name of the Kubernetes cluster that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the Kubernetes clusters within an Alibaba Cloud account.

External Configurations	None
Solution	Check whether exceptions have occurred on the Kubernetes cluster where an excessive number of error events occur.
Prerequisites	The **K8s Event Center** switch next to Kubernetes is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.14. Security of RDS instances

This topic describes the alert rules for the security of RDS instances. You can configure and enable alert rules in the Log Service console to monitor the security of RDS instances. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- RDS Slow SQL detection
- RDS Data Mass Deletion Alert
- Detection of RDS Visit through Internet
- RDS Query SQL Average Execution Time Monitoring
- RDS Instance Update Peak Monitoring
- RDS Instance Query Peak Monitoring
- RDS Instance Released Alert
- RDS Frequent Visit IP Detection

- RDS Update SQL Average Execution Time Monitoring
- Too Many RDS Login Failures Alert
- Rds Mass Data Update Event Alert
- RDS Dangerous SQL Execution Alert
- Too Many RDS SQL Execution Errors Alert

RDS Slow SQL detection

ID	sls_app_audit_db_at_rds_slow_sql
Name	RDS Slow SQL detection
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors slow SQL queries in RDS instances. If the time to execute an SQL query exceeds the value of the Threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Slow SQL detection. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the time of SQL queries. If the time of an SQL query exceeds the specified threshold, the query is a slow query. Default value: 5000. Unit: microseconds.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.

Solution	Check whether slow SQL queries occur in the RDS database that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Data Mass Deletion Alert

ID	sls_app_audit_db_at_rds_batch_del_sql
Name	RDS Data Mass Deletion Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors whether a large amount of data is deleted in RDS databases. If the number of data rows that are deleted is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Data Mass Deletion Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the maximum number of data rows that can be deleted. Default value: 10.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.

Solution	Check whether a large amount of data is deleted in the RDS database that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Detection of RDS Visit through Internet

ID	sls_app_audit_db_at_rds_internet_access
Name	Detection of RDS Visit through Internet
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors whether RDS instances are accessed by external IP addresses. If an RDS instance is accessed by an external IP address, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	You can specify a whitelist. If an RDS instance is in the whitelist and the RDS instance is accessed by an external IP address, no alert is triggered.
Solution	Do not allow RDS instances that are not included in the whitelist to be accessed by external IP addresses.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Query SQL Average Execution Time Monitoring

ID	sls_app_audit_db_at_rds_select_speed
Name	RDS Query SQL Average Execution Time Monitoring
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security

Usage	Monitors the average execution duration of an SQL query in RDS instances. If the average execution duration of an SQL query is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Query SQL Average Execution Time Monitoring. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The maximum average duration in which an SQL query statement is executed. Default value: 0.005. Unit: seconds.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value: `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the RDS database that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance Update Peak Monitoring

ID	sls_app_audit_db_at_rds_update_peak
Name	RDS Instance Update Peak Monitoring
Version	1

Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the data change in an RDS database. If the amount of data that is changed is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Instance Update Peak Monitoring. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8. • Threshold: The threshold for the maximum data amount that can be changed in an RDS database. Default value: 100. Unit: Rows. • Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored. • Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance Query Peak Monitoring

ID	sls_app_audit_db_at_rds_query_peak
Name	RDS Instance Query Peak Monitoring

Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the maximum rows of data to query each time. If the data rows that are queried is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Instance Query Peak Monitoring. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8. • Threshold: The threshold for the maximum rows of data to query each time in an RDS database. Default value: 1000. Unit: Rows. • Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored. • Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the RDS database that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Instance Released Alert

ID	sls_app_audit_db_at_rds_query_peak
Name	RDS Instance Released Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the release of RDS instances. If an RDS instance is released, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
External Configurations	None.
Solution	Check whether an exception occurs in the RDS database that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Frequent Visit IP Detection

ID	sls_app_audit_db_at_rds_visit
Name	RDS Frequent Visit IP Detection
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the frequent access from an IP address to an RDS instance. If the time of access from an IP address to an RDS instance is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Frequent Visit IP Detection. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the maximum number of times that an IP address can access an RDS instance every 2 minutes. Default value: 30.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.

External Configurations	You can specify a whitelist of IP addresses. If an RDS instance is frequently accessed by an IP address on the whitelist, no alert is triggered.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Update SQL Average Execution Time Monitoring

ID	sls_app_audit_db_at_rds_update_speed
Name	RDS Update SQL Average Execution Time Monitoring
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the time interval to change the average execution duration of an SQL query in RDS instances. If the time interval to change the average execution duration of an SQL query is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.

Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Update SQL Average Execution Time Monitoring. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the maximum time interval to change the average execution duration of an SQL query. Default value: 0.005. Unit: Seconds.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Too Many RDS Login Failures Alert

ID	sls_app_audit_db_at_rds_login_err_cnt
Name	Too Many RDS Login Failures Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the logon failures of RDS instances. If the number of logon failures of an RDS instance within 5 minutes is greater than or equal to the value of the Threshold parameter, an alert is triggered.

Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is Too Many RDS Login Failures Alert. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8. • Threshold: The threshold for the maximum number of logon failures for an RDS instance within 5 minutes. Default value: 3. • Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Rds Mass Data Update Event Alert

ID	sls_app_audit_db_at_rds_batch_update_sql
Name	Rds Mass Data Update Event Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors whether a large amount of data is changed on RDS instances. If the number of data rows changed on an RDS instance is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.

Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is Rds Mass Data Update Event Alert. You can separate multiple IDs with vertical bars (|).
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the maximum number of data rows that can be changed. Default value: 10.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

RDS Dangerous SQL Execution Alert

ID	sls_app_audit_db_at_rds_danger_sql
Name	RDS Dangerous SQL Execution Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors invalid SQL queries for RDS instances. If an invalid SQL query is detected, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.

Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is RDS Dangerous SQL Execution Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Too Many RDS SQL Execution Errors Alert

ID	sls_app_audit_db_at_rds_sql_err_cnt
Name	Too Many RDS SQL Execution Errors Alert
Version	1
Type	Cloud Platform, Alicloud, Database Security, and RDS Security
Usage	Monitors the errors that occur when SQL queries are executed. If the number of errors that occur is greater than or equal to the value of the Max errors parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.

Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is Too Many RDS SQL Execution Errors Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8.
	• Threshold: The threshold for the maximum number of errors that can occur within 2 minutes when SQL queries are executed for an RDS instance. Default value: 10.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• RDS Instance ID: The ID of the RDS instance to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all RDS instances of the specified Alibaba Cloud account are monitored.
	• Database Name: The name of the database to be monitored. You can use regular expressions when you specify this parameter. Default value `.*`. This indicates that all databases of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs on the RDS instance that triggered the alert.
Prerequisites	The **SQL Audit Log** switch of RDS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.15. Flow security of SLB

This topic describes the alert rules for the flow security of Server Load Balancer (SLB). You can configure and enable alert rules in the Log Service console. This allows you to monitor the security of SLB instances. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- Inspection of SLB Abnormal Response Length

- Inspection of SLB Abnormal Request Length
- SLB Average Response Delay Too High-8 Alert
- SLB HTTP Access Protocol Enabled Alert
- Load Balance Access UV Anomaly Inspection
- Load Balance Access PV Anomaly Inspection

Inspection of SLB Abnormal Response Length

ID	sls_app_audit_dataflow_at_slb_resp_detc
Name	Inspection of SLB Abnormal Response Length
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Detects whether the length of SLB response is abnormal. If the number of SLB responses that have an abnormal length is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.
Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is Inspection of SLB Abnormal Response Length. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Threshold: The maximum number of SLB responses that have an abnormal length during the 4 hour window. An average response length is calculated per minute. Default value: 10. • Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application. • SLB Instance Name: The name of the SLB instance that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the SLB instances that are attached to your Alibaba Cloud accounts.
External Configurations	None
Solution	Check whether exceptions have occurred on the SLB instances that have an abnormal length in a large number of responses.

Prerequisites	The **Lay-7 Access** switch next to SLB is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations** .

Inspection of SLB Abnormal Request Length

ID	sls_app_audit_dataflow_at_slb_req_detc
Name	Inspection of SLB Abnormal Request Length
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Detects whether the length of SLB request is abnormal. If the number of SLB requests that have abnormal length is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.
Parameter Settings	• Alarm Name: The name of the alert. By default, the value of this parameter is Inspection of SLB Abnormal Request Length. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Threshold: The maximum number of SLB requests that have an abnormal length during the 4 hour window. An average request length is calculated per minute. Default value: 10. • Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application. • SLB Instance Name: The name of the SLB instance that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the SLB instances that are attached to your Alibaba Cloud accounts.
External Configurations	None
Solution	Check whether exceptions have occurred on the SLB instances that have an abnormal length in a large number of requests.

Prerequisites	The **Lay-7 Access** switch next to SLB is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

SLB Average Response Delay Too High-8 Alert

ID	sls_app_audit_dataflow_at_slb_latency
Name	SLB Average Response Delay Too High-8 Alert
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Checks whether the average response delay of Server Load Balancer (SLB) instances is too high. If the average response time of SLB instances is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alarm Name: The name of the alert. By default, the value of this parameter is SLB Average Response Delay Too High-8 Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum average response delay of the SLB instance during the 2 minute window. Default value: 0.5. Unit: seconds.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• SLB Instance Name: The name of the SLB instance that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the SLB instances that are attached to your Alibaba Cloud accounts.

External Configurations	None
Solution	Check whether exceptions have occurred on SLB instances whose average response delay is too high.

Prerequisites	The **Lay-7 Access** switch next to SLB is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

SLB HTTP Access Protocol Enabled Alert

ID	sls_app_audit_dataflow_at_slb_http
Name	SLB HTTP Access Protocol Enabled Alert
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Detects whether the Server Load Balancer (SLB) accesses the server through HTTPS protocol. When the SLB accesses the server through HTTP protocol, an alert will be triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
External Configurations	You can configure a whitelist of SLB instances for which HTTP protocol is enabled. If HTTP protocol is enabled for SLB instances on the whitelist, no alert is be triggered.
Solution	Disable HTTP protocol for the SLB instances that are not included in the whitelist.
Prerequisites	The **Operations Log** switch next to ActionTrail is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Load Balance Access UV Anomaly Inspection

ID	sls_app_audit_dataflow_at_slb_uv_detc
Name	Load Balance Access UV Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Detects the anomaly of Unique Visitors (UVs) of Server Load Balancers (SLB). If the number of UVs of abnormal access to SLB instances is greater than or equal to the value of the Threshold parameter, an alert is triggered.

Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.
Parameter Settings	• Alarm Name: The name of the alert. By default, the value of this parameter is Load Balance Access UV Anomaly Inspection. You can specify a unique name for each alert based on the metrics that you want to monitor. • Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8 • Threshold: The maximum number of UVs of abnormal access during the 4 hour window. One UV value is calculated per minute. Default value: 10. • Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application. • SLB Instance Name: The name of the SLB instance that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the SLB instances that are attached to your Alibaba Cloud accounts.
External Configurations	None
Solution	Check whether exceptions have occurred on the SLB instances whose UVs of abnormal access are in a large number.
Prerequisites	The **Lay-7 Access** switch next to SLB is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Load Balance Access PV Anomaly Inspection

ID	sls_app_audit_dataflow_at_slb_pv_detc
Alert Name	Load Balance Access PV Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Flow Security, SLB Flow Security
Usage	Detects excessive number of page views (PVs) of Server Load Balancer (SLB) instances. If the number of PVs of abnormal access to SLB instances is greater than or equal to the value of the Threshold parameter, an alert is triggered.

Check Frequency	Fixed interval: 4 hours.
TimeRange	The data of the last 4 hours is checked.

Parameter Settings	• Alarm Name: The name of the alert. By default, the value of this parameter is Server Load Balancer access UV anomaly detection. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum number of PVs of abnormal access during the 4 hour window. One PV value is calculated per minute. Default value: 10.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`, which indicates the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• SLB Instance Name: The name of the SLB instance that you want to monitor. Regular expressions are supported. The default value is `.*`, which indicates the SLB instances that are attached to your Alibaba Cloud accounts.

External Configurations	None
Solution	Check whether exceptions have occurred on the SLB instances whose PVs of abnormal access are in a large number.
Prerequisites	The **Lay-7 Access** switch next to SLB is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.16. Flow security of API Gateway

This topic describes the alert rules for traffic security of API Gateway. You can configure and enable alert rules in the Log Service console. This allows you to monitor the flow security of API Gateway. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- APIgateway Server Average Delay Too High-8 Alert
- APIGateway Backend Server Error Rate Too High-8 Alert
- APIgateway Request Success Rate Too Low Alert

APIgateway Server Average Delay Too High-8 Alert

ID	sls_app_audit_dataflow_at_api_latency
Name	APIgateway Server Average Delay Too High-8 Alert
Version	1
Type	Cloud Platform, Alicloud, Flow Security, APIGateway Flow Security
Usage	Monitors the average server-side delay of API requests in API Gateway. If the average server-side delay of API requests is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is APIgateway Server Average Delay Too High-8 Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum average server-side delay of API requests during the 2 minute window. Default value: 100. Unit: milliseconds.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• API Name: The name of the API to be monitored. Regular expressions are supported. The default value is `.*`. This is used to identify all APIs.

External Configurations	None
Solution	Check whether exceptions have occurred on the API requests whose average server-side delay is too high.
Prerequisites	The **Access Log** switch next to API Gateway instance is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

APIGateway Backend Server Error Rate Too High-8 Alert

ID	sls_app_audit_dataflow_at_api_err_rate
Name	APIGateway Backend Server Error Rate Too High-8 Alert
Version	1
Type	Cloud Platform, Alicloud, Flow Security, APIGateway Flow Security
Usage	Monitors the error rate of API requests at the backend server in API Gateway. If the error rate of API requests at the backend server in API Gateway is greater than or equal to the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is APIGateway Backend Server Error Rate Too High-8 Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The maximum error rate of API requests at the backend during the 2 minute window. Default value: 0.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`. This is used to identify all the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• API name: The name of the API to be monitored. Regular expressions are supported. The default value is `.*`. This is used to identify all all APIs.

External Configurations	None
Solution	Check whether exceptions have occurred on the APIs whose error rates at the server end are too high.
Prerequisites	The **Access Log** switch next to API Gateway instance is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

APIgateway Request Success Rate Too Low Alert

ID	sls_app_audit_dataflow_at_api_req_rate
Name	APIgateway Request Success Rate Too Low Alert
Version	1
Type	Cloud Platform, Alicloud, Flow Security, APIGateway Flow Security
Usage	Monitors the request success rate of API requests in an API gateway. If the success rate of API requests in an API gateway is lower than the value of the Threshold parameter, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	• Alert Name: The name of the alert. By default, the value of this parameter is APIgateway Request Success Rate Too Low Alert. You can specify a unique name for each alert based on the metrics that you want to monitor.
	• Severity: Critical-10, High-8, Medium-6, Low-4, and Report-2. Default value: High-8
	• Threshold: The minimum success rate of API requests in an API gateway. Default value: 95%.
	• Account ID (Aliuid): The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates that the Alibaba Cloud accounts that start with 156133.
	  ◦ The default value is `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.
	• API Name: The name of the API to be monitored. Regular expressions are supported. The default value is `.*`. This is used to identify all APIs.

External Configurations	None
Solution	Check whether exceptions have occurred on the APIs whose success rates of requests are too low.
Prerequisites	The **Access Log** switch next to API Gateway instance is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.17. Security of OSS traffic

This topic describes the alert rules for the security of Object Storage Service (OSS) traffic. You can configure and enable alert rules in the Log Service console to monitor the security of OSS traffic. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- OSS Flow Anomaly Inspection
- OSS Inflow Anomaly Inspection
- OSS Outflow Anomaly Inspection
- OSS Access PV Anomaly Inspection
- OSS Access UV Anomaly Inspection
- OSS Bucket Valid Request Rate Too Low Alert
- Detection of OSS Bucket Visit through Internet

OSS Flow Anomaly Inspection

ID	sls_app_audit_dataflow_at_oss_flow_detc
Name	OSS Flow Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.
Usage	Monitors the inbound and outbound traffic of OSS. If the number of traffic exceptions exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.

Parameters Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of OSS traffic exceptions. Default value: 10. If the number of traffic exceptions exceeds the threshold within 4 hours, an alert is triggered. A traffic value is calculated every minute. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Inflow Anomaly Inspection

ID	sls_app_audit_dataflow_at_oss_inflow_detc
Alert Name	OSS Inflow Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.
Usage	Monitors the inbound traffic of OSS. If the number of traffic exceptions exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.

Time Range	The data of the last 4 hours is checked.
Parameters Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of inbound traffic exceptions of OSS. Default value: 10. If the number exceeds the threshold within 4 hours, an alert is triggered. An inbound traffic value is calculated every minute. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Outflow Anomaly Inspection

ID	sls_app_audit_dataflow_at_oss_outflow_detc
Name	OSS Outflow Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.
Usage	Monitors the outbound traffic of OSS. If the number of outbound traffic exceptions exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.

Time Range	The data of the last 4 hours is checked.
Parameter Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of outbound traffic exceptions of OSS. Default value: 10. If the number of traffic exceptions exceeds the threshold within 4 hours, an alert is triggered. A traffic value is calculated every minute. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Access PV Anomaly Inspection

ID	sls_app_audit_dataflow_at_oss_pv_detc
Name	OSS Access PV Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.
Usage	Monitors the PVs of OSS. If the number of PV exceptions exceeds the specified threshold, an alert is triggered.

Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.
Parameters Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of PV exceptions of OSS. Default value: 10. If the number of PV exceptions exceeds the threshold within 4 hours, an alert is triggered. A PV value is calculated every minute. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Access UV Anomaly Inspection

ID	sls_app_audit_dataflow_at_oss_uv_detc
Name	OSS Access UV Anomaly Inspection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.

Usage	Monitors the UVs of OSS. If the number of UV exceptions exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 hours.
Time Range	The data of the last 4 hours is checked.
Parameters Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of UV exceptions of OSS. Default value: 10. If the number of UV exceptions exceeds the threshold within 4 hours, an alert is triggered. A value is calculated every minute. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Bucket Valid Request Rate Too Low Alert

ID	sls_app_audit_dataflow_at_oss_req_rate
Name	OSS Bucket Valid Request Rate Too Low Alert
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.

Usage	Monitors the valid request rate of OSS buckets. If the rate is lower than the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameters Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the valid request rate for OSS buckets. Default value: 95. If the rate is lower than the threshold, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Detection of OSS Bucket Visit through Internet

ID	sls_app_audit_dataflow_at_oss_internet_access
Name	Detection of OSS Bucket Visit through Internet
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Flow Security.

Usage	Monitors the access of OSS buckets over the Internet. If an OSS bucket is accessed over the Internet, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	You can specify the following parameters: **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
External Configurations	You can specify a whitelist of accounts. If an OSS bucket belongs to an account on the whitelist and the OSS bucket is accessed over the Internet, no alert is triggered.
Solution	Do not allow OSS buckets that do not belong to an account on the whitelist to be accessed over the Internet.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.18. The security of Kubernetes traffic

This topic describes the alert rules for the security of Kubernetes traffic. You can configure and enable alert rules in the Log Service console to monitor the security of Kubernetes traffic.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- K8s Ingress Response Delay Too High Alert
- K8s Ingress Request Success Rate Too Low Alert
- K8s Ingress Average Request Latency Too High Alert
- Too Many K8s Illegal Access Alert

K8s Ingress Response Delay Too High Alert

ID	sls_app_audit_dataflow_at_ingress_resp
Name	K8s Ingress Response Delay Too High Alert
Version	1
Type	Cloud Platform, Alicloud, Data Security, and K8s Flow Security
Usage	Monitors the average backend response latency of Kubernetes Ingress. If the latency is higher than the specified threshold, an alert is triggered.

Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the average backend response latency of Kubernetes Ingress. Default value: 500. Units: milliseconds. If the average latency exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **K8s Cluster Name**: The name of the Kubernetes cluster to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates all Kubernetes clusters of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the Kubernetes cluster that triggered the alert.
Prerequisites	The **Ingress Log** switch of Kubernetes is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

K8s Ingress Average Request Latency Too High Alert

ID	sls_app_audit_dataflow_at_ingress_latency
Name	K8s Ingress Average Request Latency Too High Alert
Version	1
Type	Cloud Platform, Alicloud, Data Security, and K8s Flow Security

Usage	Monitors the average request latency of Kubernetes Ingress. If the latency is higher than the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the average request latency of Kubernetes Ingress. Default value: 200. Units: milliseconds. If the average latency exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **K8s Cluster Name**: The name of the Kubernetes cluster to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates all Kubernetes clusters of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the Kubernetes cluster that triggered the alert.
Prerequisites	The **Ingress Log** switch of Kubernetes is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

K8s Ingress Request Success Rate Too Low Alert

ID	sls_app_audit_dataflow_at_ingress_rate
Name	K8s Ingress Request Success Rate Too Low Alert
Version	1

Type	Cloud Platform, Alicloud, Data Security, and K8s Flow Security
Usage	Monitors the request success rate of Kubernetes Ingress. If the rate is lower than the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the request success rate of Kubernetes Ingress. Default value: 90%. If the request success rate is lower than the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **K8s Cluster Name**: The name of the Kubernetes cluster to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates all Kubernetes clusters of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the Kubernetes cluster that triggered the alert.
Prerequisites	The **Ingress Log** switch of Kubernetes is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Too Many K8s Illegal Access Alert

ID	sls_app_audit_dataflow_at_k8s_visit
Name	Too Many K8s Illegal Access Alert
Version	1

Type	Cloud Platform, Alicloud, Data Security, and K8s Flow Security
Usage	Monitors the access to Kubernetes clusters. If the number of invalid access to a Kubernetes cluster exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameters Settings	You can specify the following parameters: • **Alert Name**: The name of the alert. You can create multiple alerts. • **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of invalid access to a Kubernetes cluster. Default value: 3. If the number of invalid access to a Kubernetes cluster exceeds the threshold within 2 minutes, an alert is triggered. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter. ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored. ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored. • **K8s Cluster Name**: The name of the Kubernetes cluster to be monitored. You can use regular expressions when you specify this parameter. ◦ You can also use wildcards for the regular expressions, such as `.*`. ◦ Default value: `.*`. This indicates all Kubernetes clusters of the specified Alibaba Cloud account are monitored.
External Configurations	None.
Solution	Check whether an exception occurs in the Kubernetes cluster that triggered the alert.
Prerequisites	The **Ingress Log** switch of Kubernetes is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.19. Security of OSS data

This topic describes the alert rules for the security of OSS data. You can configure and enable alert rules in the Log Service console to monitor the security of OSS data. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- OSS Object Frequent Deletion Alert
- OSS Bucket Account Access Control

OSS Object Frequent Deletion Alert

ID	sls_app_audit_storage_at_oss_obj_del
Name	OSS Object Frequent Deletion Alert
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Data Security
Usage	Monitors the delete operations in OSS buckets. If the delete operations exceed the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of delete operations. Default value:10. If the number of delete operations in an OSS bucket exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Bucket Name**: The name of the OSS bucket to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all OSS buckets of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the OSS bucket that triggered the alert.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

OSS Bucket Account Access Control

ID	sls_app_audit_storage_at_oss_access_control
Name	OSS Bucket Account Access Control
Version	1
Type	Cloud Platform, Alicloud, Data Security, and OSS Data Security
Usage	Monitors the access to OSS bucket. If an OSS bucket is accessed by an unspecified Alibaba Cloud account or RAM user, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	You can specify the following parameters: **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
External Configurations	You can specify a whitelist and add the Alibaba Cloud account and RAM user to the whitelist. If an OSS bucket is accessed by a whitelist account, no alert is triggered.
Solution	Do not allow Alibaba Cloud accounts or RAM users that are not included in the whitelist to access OSS buckets.
Prerequisites	The **Access Log** switch of OSS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.20. Data security of NAS

This topic describes the alerts for the data security of Apsara File Storage NAS (NAS). You can set and then enable alert instances to trigger alerts in a timely manner. In this case, you can identify if errors exist in NAS.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- NAS Error Operation Detection
- NAS Mass Deletion Alert

NAS Error Operation Detection

ID	sls_app_audit_storage_at_nas_err_op
Name	NAS Error Operation Detection
Version	1
Type	Cloud Platform, Alicloud, Data Security, and NAS Data Security
Usage	Monitors the error operations on NAS volumes. If the number of error operations on a NAS volume exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of error operations. Default value: 5. If the number of error operations on a NAS volume exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Volume**: The name of the volume to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates all the volumes of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the NAS volume that triggered the alert.

Prerequisites	The **Access Log** switch of NAS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

NAS Mass Deletion Alert

ID	sls_app_audit_storage_at_nas_file_del
Name	NAS Mass Deletion Alert
Version	1
Type	Cloud Platform, Alicloud, Data Security, and NAS Data Security
Usage	Monitors the delete operations on NAS volumes. If the number of delete operations on a NAS volume exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of delete operations. If the number of delete operations on a NAS volume exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Volume**: The name of the volume to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates all the volumes of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the NAS volume that triggered the alert.

Prerequisites	The **Access Log** switch of NAS is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.21. Security events of WAF

This topic describes the alerts for the security events of Web Application Firewall (WAF). You can configure and enable alert rules in the Log Service console to monitor the security events of WAF. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- Too Many Attacks on Hosts Protected by WAF Alert
- Application Firewall Valid Request Rate Too Low Alert

Too Many Attacks on Hosts Protected by WAF Alert

ID	sls_app_audit_secure_at_waf_attack
Name	Too Many Attacks on Hosts Protected by WAF Alert
Version	1
Type	Cloud Platform, Alicloud, Security Event, and WAF Security Event
Usage	Monitors website attacks. If the number of attacks on a website that is protected by WAF exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of attacks on a website. Default value: 5. If the number exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Hosts**: The name of the website to be monitored.
	  ◦ You can use regular expressions when you specify this parameter. You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all websites protected by WAF of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the website that triggered the alert.
Prerequisites	The **Access Log** switch of Web Application Firewall is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Application Firewall Valid Request Rate Too Low Alert

ID	sls_app_audit_secure_at_waf_rate
Name	Application Firewall Valid Request Rate Too Low Alert
Version	1
Type	Cloud Platform, Alicloud, Security Event, and WAF Security Event
Usage	Monitors the valid request rate to a website WAF blocks and filters inbound traffic to your website. If the valid request rate to a website is lower than the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the valid request rate to a website. Default value: 90%. If the rate is lower than the specified threshold, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Hosts**: The name of the website to be monitored.
	  ◦ You can use regular expressions when you specify this parameter. You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all websites protected by WAF of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the website that triggered the alert.
Prerequisites	The **Access Log** switch of Web Application Firewall is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.8.2.22. TDI security events

This topic describes the alert rules for the security events of threat detection and identification (TDI). You can configure and enable alert rules in the Log Service console. This allows you to monitor the security events of TDI. If an alert is triggered, you can identify the cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other related operations, see Configure alerts.

- Cloud Security Center Request Success Rate Too Low
- Cloud Security Center Valid Request Rate Too Low Alert
- Too Many New Alarms In Cloud Security Center
- Too Many New Vulnerabilities In Cloud Security Centers

- Too Many High-Priority Alarms In Cloud Security Center

Cloud Security Center Request Success Rate Too Low

ID	sls_app_audit_secure_at_sas_dns_rate
Name	Cloud Security Center Request Success Rate Too Low
Version	1
Type	Cloud Platform, Alicloud, Security Event, TDI Security Event
Usage	Monitors the success rate of DNS requests sent to Security Center If the success rate of DNS requests sent to Security Center is lower than this threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	The following rules describe the parameter settings of the alert: • **Alert Name**: The name of the alert. You can create multiple alert instances. • **Severity**: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold success rate of DNS requests sent to Security Center. Default value: 90%. If the success rate of DNS request sent to Security Center is lower than the specified threshold during the 2 minute window, an alert is triggered. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ Default value: `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.
External Configurations	None
Solution	Check whether exceptions have occurred on DNS requests that are sent to Security Center.
Prerequisites	The switch next to Security Center(SAS) is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Cloud Security Center Valid Request Rate Too Low Alert

ID	sls_app_audit_secure_at_sas_rate

Name	Cloud Security Center Valid Request Rate Too Low Alert
Version	1
Type	Cloud Platform, Alicloud, Security Event, TDI Security Event
Usage	Monitors the rate of valid requests sent to Security Center. If the rate of valid requests sent to the website is lower than the specified threshold after all requests are filtered by Security Center, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.
Parameter Settings	The following rules describe the parameter settings of the alert: • **Alert Name**: The name of the alert. You can create multiple alert instances. • **Severity**: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold rate of the valid request to the website. Default value: 90%. If the rate of valid request sent to the website is lower than the threshold after all requests are filtered by Security Center protection, an alert is triggered. This applies during the last 2 minutes when the requests are filtered. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ Default value: `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application. • **Website (host)**: The name of the website that you want to monitor. Regular expressions are supported. ◦ You can use regular expressions, such as `.*` to make configurations. ◦ Default value: `.*`. This is used to identify all websites within the Alibaba Cloud account.
External Configurations	None
Solution	Check whether exceptions have occurred on the request events that are sent to Security Center. You can also check whether a large number of attack events have occurred.
Prerequisites	The switch next to Security Center(SAS) is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Too Many New Alarms In Cloud Security Center

ID	sls_app_audit_secure_at_sas_new_alert
Name	Too Many New Alarms In Cloud Security Center
Version	1
Type	Cloud Platform, Alicloud, Security Event, TDI Security Event
Usage	Monitors the number of new alerts in Security Center. If the number of new alerts in Security Center exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.
Parameter Settings	The following rules describe the parameter settings of the alert: • **Alert Name**: The name of the alert. You can create multiple alert instances. • **Severity**: Critical-10, High-8, Medium-6, Low-4, and Report-2. • **Threshold**: The threshold for the number of new alerts. Default value: 2. If the number of new alerts in Security Center exceeds the specified threshold during the 5 minute window, an alert is triggered. • **Account ID (Aliuid)**: The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported. ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133. ◦ Default value: `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.
External Configurations	None
Solution	Check the new alerts in Security Center.
Prerequisites	The switch next to Security Center(SAS) is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Too Many New Vulnerabilities In Cloud Security Centers

ID	sls_app_audit_secure_at_sas_new_vul
Name	Too Many New Vulnerabilities In Cloud Security Centers
Version	1

Type	Cloud Platform, Alicloud, Security Event, TDI Security Event
Usage	Monitors the number of new vulnerabilities in Security Center. If the number of new vulnerabilities in Security Center exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.

Parameter Settings	The following rules describe the parameter settings of the alert:
	• **Alert Name**: The name of the alert. You can create multiple alert instances.
	• **Severity**: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of new vulnerabilities. Default value: 1. If the number of new vulnerabilities in Security Center exceeds the specified threshold during the 5 minute window, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ◦ Default value: `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.

External Configurations	None
Solution	Check the new vulnerabilities in Security Center.
Prerequisites	The switch next to Security Center(SAS) is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

Too Many High-Priority Alarms In Cloud Security Center

ID	sls_app_audit_secure_at_sas_ser_alert
Name	Too Many High-Priority Alarms In Cloud Security Center
Version	1
Type	Cloud Platform, Alicloud, Security Event, TDI Security Event
Usage	Monitors the number of high-priority alerts in Security Center. If the number of high-priority alerts in Security Center exceeds the specified threshold, an alert is triggered.

Check Frequency	Fixed interval: 4 minutes.
Time Range	The data of the last 5 minutes is checked.

Parameter Settings	The following rules describe the parameter settings of the alert:
	• **Alert Name**: The name of the alert. You can create multiple alert instances.
	• **Severity**: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of high-priority alerts. Default value: 1. If the number of high-priority alerts in Security Center exceeds this threshold, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account related to the API gateway that you want to monitor. Regular expressions are supported.
	  ○ You can separate multiple IDs with vertical bars (|). You can also use regular expressions `.*` in the IDs. For example, 156133.* indicates the Alibaba Cloud accounts that start with 156133.
	  ○ Default value: `.*`. This is used to identify the Alibaba Cloud accounts that are configured in the Log Audit Service application.

External Configurations	None
Solution	You can monitor the high-priority alert in Security Center.
Prerequisites	The switch next to Security Center(SAS) is turned on. To turn on the switch, go to the Log Audit Service page, and then choose **Audit Configurations > Access to Cloud Products > Global Configurations**.

1.8.2.23. Security events of Cloud Firewall

This topic describes the alert rules for the security events of Cloud Firewall. You can configure and enable alert rules in the Log Service console to trigger alerts to monitor the security events of Cloud Firewall. If an alert is triggered, you can identify the error cause and fix the error at the earliest opportunity.

Alert rules

The following alert rules are supported. For information about how to set alert parameters, configure whitelists, and perform other relevant operations, see Configure alerts.

- Cloudfirewall Inflow Block Alarm
- Cloudfirewall Outflow Block Alert

Cloudfirewall Inflow Block Alarm

ID	sls_app_audit_secure_at_cfw_in_block
Name	Cloudfirewall Inflow Block Alarm

Version	1
Type	Cloud Platform, Alicloud, Security Event, and Cloudfirewall Security Event
Usage	Monitors the inbound traffic that is intercepted by Cloud Firewall. If the number of inbound traffic interception for an access protocol exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of inbound traffic interception. Default value: 10. If the number exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Access Protocol Name**: The name of the access protocol to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all access protocols of the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the inbound traffic that is intercepted by the Cloud Firewall.
Prerequisites	The **Internet Access Log** switch of Cloud Firewall is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

Cloudfirewall Outflow Block Alert

ID	sls_app_audit_secure_at_cfw_out_block

Name	Cloudfirewall Outflow Block Alert
Version	1
Type	Cloud Platform, Alicloud, Security Event, and Cloudfirewall Security Event
Usage	Monitors the outbound traffic that is intercepted by Cloud Firewall. If the number of outbound traffic interceptions for an access protocol exceeds the specified threshold, an alert is triggered.
Check Frequency	Fixed interval: 1 minute.
Time Range	The data of the last 2 minutes is checked.

Parameter Settings	You can specify the following parameters:
	• **Alert Name**: The name of the alert. You can create multiple alerts.
	• **Severity**: The severity level of the alert. Valid values: Critical-10, High-8, Medium-6, Low-4, and Report-2.
	• **Threshold**: The threshold for the number of times outbound traffic is intercepted. Default value: 10. If the number exceeds the threshold within 2 minutes, an alert is triggered.
	• **Account ID (Aliuid)**: The ID of the Alibaba Cloud account that you want to monitor. You can use regular expressions when you specify this parameter.
	  ◦ You can separate multiple IDs with vertical bars (|). You can also use wildcards for the regular expressions, such as `.*`. For example, 156133.* indicates that Alibaba Cloud accounts that start with 156133 are monitored.
	  ◦ Default value: `.*`. This indicates all Alibaba Cloud accounts configured in the Log Audit Service application are monitored.
	• **Access Protocol Name**: The name of the access protocol to be monitored. You can use regular expressions when you specify this parameter.
	  ◦ You can also use wildcards for the regular expressions, such as `.*`.
	  ◦ Default value: `.*`. This indicates that all access protocols under the specified Alibaba Cloud account are monitored.

External Configurations	None.
Solution	Check whether an exception occurs in the outbound traffic that is intercepted by the Cloud Firewall.
Prerequisites	The **Internet Access Log** switch of Cloud Firewall is turned on. To turn on the switch, go to the Log Audit Service console, and then choose **Log Audit Service > Access to Cloud Products > Global Configurations**.

1.9. Use Terraform to configure Log Audit Service

This topic describes how to use Terraform and its CLI to configure Log Audit Service.

Prerequisites

Terraform is installed and configured. For more information, see Use Terraform in Cloud Shell and Install and configure Terraform in the local PC.

Context

Terraform is an open source tool that you can use to preview, configure, and manage the infrastructure and resources of cloud services in a secure and efficient manner. Terraform provides an easy-to-use CLI that allows you to deploy configuration files on the workloads of Alibaba Cloud services or third-party cloud services and manage the versions of the configuration files.

Alibaba Cloud supports more than 163 resources and 113 data sources across multiple Alibaba Cloud services in the following categories: computing, storage, networking, CDN, container, middleware, and database. This helps a large number of customers migrate data to the cloud in an automated manner. For more information, see Alibaba Cloud Provider.

Benefits of Terraform

- Multi-cloud infrastructure deployment

  Terraform is suitable for multi-cloud scenarios in which multiple similar infrastructures are deployed across Alibaba Cloud, third-party cloud services, and data centers. Terraform allows you to use the same tools and similar configuration files to manage infrastructures across different cloud service providers.

- Automated infrastructure management

  Terraform allows you to create configuration file templates to define, provision, and configure Elastic Compute Service (ECS) resources in a repeated and predictable manner. This reduces human errors during deployment and management operations. You can use the same template multiple times to create identical development, test, and production environments.

- Infrastructure as code (IaC)

  Terraform supports the code-based management and maintenance of resources. Terraform stores a copy of the current configurations of your infrastructure. This way, you can track changes made to the components in the IaC system and share infrastructure configurations with other users.

- Reduced development costs

  You can use Terraform to create development and deployment environments based on your business requirements. This helps you reduce development and deployment costs. In addition, you can use Terraform to evaluate development costs before you make changes to your system.

Step 1: Specify the identity information and region of the central project for Log Audit Service

Use environment variables to specify the identity information and region of the central project for Log Audit Service.

```
export ALICLOUD_ACCESS_KEY="AccessKey ID"
export ALICLOUD_SECRET_KEY="AccessKey Secret"
export ALICLOUD_REGION="cn-huhehaote"
```

Parameter	Description
ALICLOUD_ACCESS_KEY	The AccessKey ID of your Alibaba Cloud account. For more information, see AccessKey pair.
ALICLOUD_SECRET_KEY	The AccessKey secret of your Alibaba Cloud account. For more information, see AccessKey pair.
ALICLOUD_REGION	The region where the central project of Log Audit Service resides. The following regions are supported: • Chinese mainland: China (Qingdao), China (Beijing), China (Hohhot), China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Hong Kong) • Outside the Chinese mainland: Singapore (Singapore), Japan (Tokyo), Germany (Frankfurt), and Indonesia (Jakarta)

Step 2: Complete RAM authorization

If the AliyunServiceRoleForSLSAudit service-linked role does not exist in the central account, you must first create the service-linked role. For more information, see Initially configure Log Audit Service.

For information about how to configure other member accounts in custom authentication mode and the related custom policies, see Use a custom policy to authorize Log Service to collect and synchronize logs.

Step 3: Configure Log Audit Service

1. Create a Terraform directory named *sls* and create a file named *terraform.tf* in the directory.
2. Open the *terraform.tf* file and add the following content:

   ```
   resource "alicloud_log_audit" "example" {
     display_name = "tf-audit-test"
     aliuid       = "1379186349****"
   }
   ```

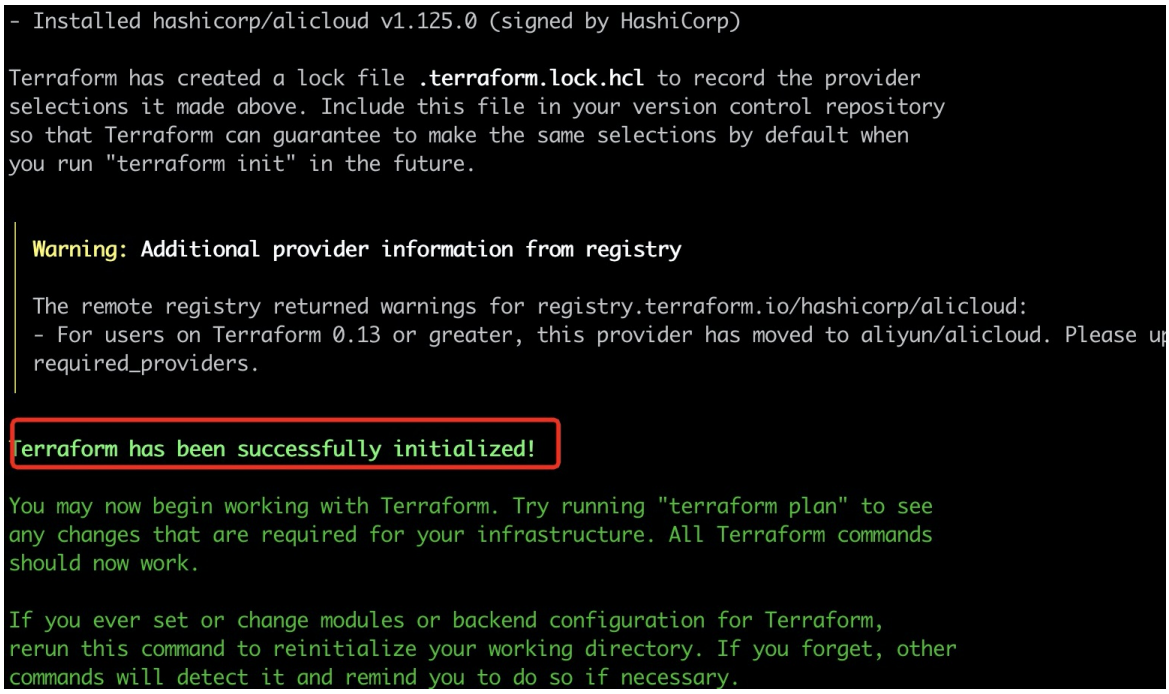
   The following table describes the parameters.

Parameter	Description
example	The name of the resource. You can specify a custom name.
display_name	The name of the collection configuration. You can specify a custom name.
aliuid	The ID of your Alibaba Cloud account.

3. Run the following command in the *sls* directory to initialize the directory:

   ```
   terraform init
   ```

If the command output contains `Terraform has been successfully initialized!`, the directory is initialized.



4. Open the *terraform.tf* file and configure the parameters of Log Audit Service.

   The following sample code provides configuration examples. For more information about the parameters, see alicloud_log_audit.

   - Single-account logging

     ```
     resource "alicloud_log_audit" "example" {
       display_name = "tf-audit-test"
       aliuid       = "1379186349****"
       variable_map = {
         "actiontrail_enabled" = "true",
         "actiontrail_ttl" = "180"
       }
     }
     ```

   - Multi-account logging

     You can configure the multi-account logging feature in custom authentication mode or resource directory mode. In custom authentication mode, the central account is an Alibaba Cloud account. In resource directory mode, the central account must be a management account or a delegated administrator account of Resource Directory. For more information, see Configure multi-account collection.

- Custom authentication mode

```
resource "alicloud_log_audit" "example" {
  display_name = "tf-audit-test"
  aliuid     = "1379186349****"
  variable_map = {
    "actiontrail_enabled" = "true",
    "actiontrail_ttl" = "180"
  }
  multi_account = ["1257918632****", "1324567349****"]
}
```

- Custom mode in resource directory mode

```
resource "alicloud_log_audit" "example" {
  display_name = "tf-audit-test"
  aliuid    = "1379186349****"
  variable_map = {
    "actiontrail_enabled" = "true",
    "actiontrail_ttl" = "180"
  }
  multi_account = ["1257918632****", "1324567349****"]
resource_directory_type="custom"
}
```

- All Members mode in resource directory mode

```
resource "alicloud_log_audit" "example" {
  display_name = "tf-audit-test"
  aliuid       = "1379186349****"
  variable_map = {
    "actiontrail_enabled" = "true",
    "actiontrail_ttl" = "180"
  }
resource_directory_type="all"
}
```

The following table describes the parameters.

Parameter	Description

Parameter	Description
multi_account	If you configure multi-account logging in custom authentication mode or by using the Custom mode in resource directory mode, you must configure the multi_account parameter. **Note** The custom authentication mode requires complex configurations. We recommend that you configure multi-account logging in resource directory mode. ◦ If you use the custom authentication mode, the resource_directory_type parameter is unavailable, and you must set the multi_account parameter to the ID of an Alibaba Cloud account. ◦ If you use the Custom mode in resource directory mode, the resource_directory_type parameter is set to custom, and you must set the multi_account parameter to a member in your resource directory.
resource_directory_type	If you configure multi-account logging in resource directory mode, you must configure the resource_directory_type parameter. Valid values: ◦ *all*: The All Members mode in resource directory mode is used. ◦ *custom*: The Custom mode in resource directory mode is used. **Note** If you use the custom authentication mode, you do not need to configure the resource_directory_type parameter.
variable_map	Specifies the objects to collect, whether to collect specific data, and the retention period of the objects. For information about the parameters in the variable_map parameter, see Appendix: parameters in variable_map.

5. Apply the configurations in the *terraform.tf* file.
   i. Run the following command:

```
terraform apply
```

ii. Enter *yes*.

If the command output contains `Apply complete!`, the configurations are in effect and Log Audit Service collects and stores logs based on the configurations.

```
Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

alicloud_log_audit.example: Creating...
alicloud_log_audit.example: Creation complete after 3s [id=tf-audit-test]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

Related operations

You can use Terraform to perform the following operations:

- Import existing collection configurations.

```
terraform import alicloud_log_audit.example tf-audit-test
```

You must replace *example* and *tf-audit-test* with the actual values.

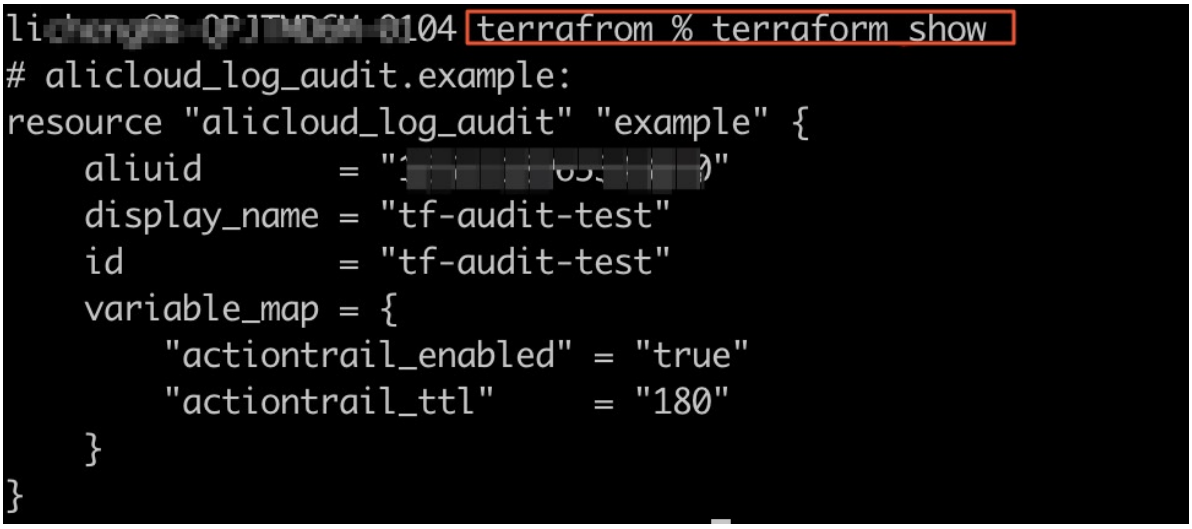
```
lichang@B-QPJTMD6M-0104 terrafrom % terraform import alicloud_log_audit.example tf-audit-test
alicloud_log_audit.example: Importing from ID "tf-audit-test"...
alicloud_log_audit.example: Import prepared!
  Prepared alicloud_log_audit for import
alicloud_log_audit.example: Refreshing state... [id=tf-audit-test]

Import successful!

The resources that were imported are shown above. These resources are now in
your Terraform state and will henceforth be managed by Terraform.
```

After the command is run, you can view the content of the *terraform.tfstate* file in the Terraform directory. The *terraform.tfstate* file contains the imported collection configurations.

> **Notice**
>
> - If you want to migrate the imported collection configurations to the *terraform.tf* file, you must copy the configurations and adjust the format of the configurations to meet the format requirements of the *terraform.tf* file.
> - If you run the **terrraform apply** or **terraform import** command once in the Terraform directory, the next execution of the **terraform import** command fails. Before you can run the **terraform import** command again, you must delete the *terraform.tfstate* file from the directory.

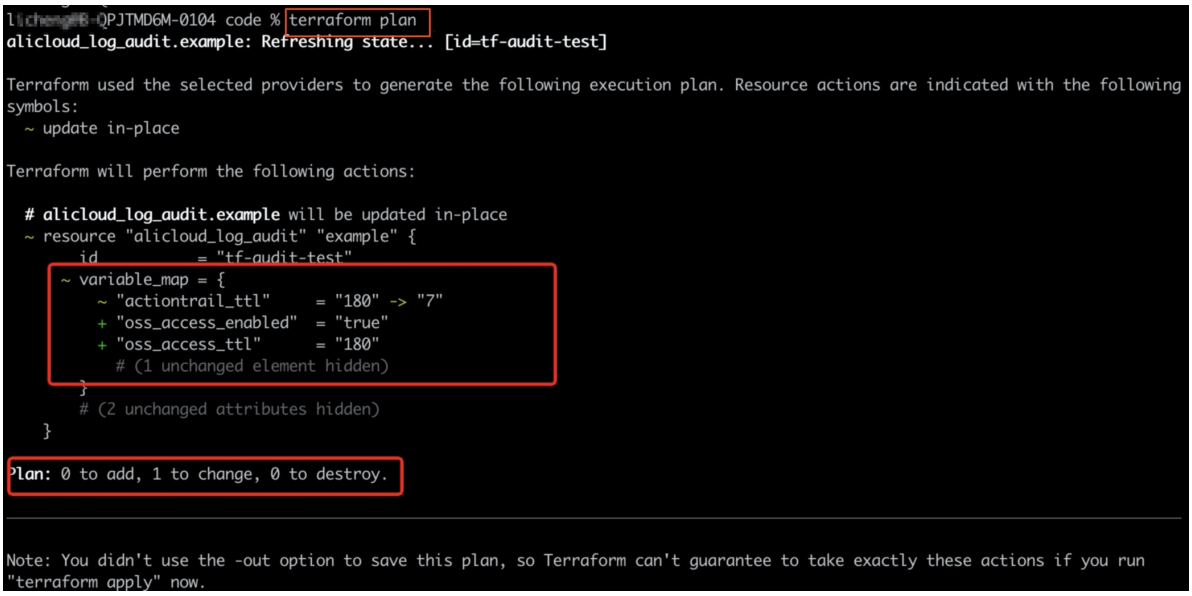
- View the current collection configurations.

```
terraform show
```



- View the differences between the *terraform.tf* file in the Terraform directory and the collection configurations that are in effect.

```
terraform plan
```



Example

If you use Terraform to configure collection policies in Log Audit Service, take note of the configuration of special character escapes and multi-line policies. For example, if you want to collect logs from only virtual private clouds (VPCs) whose tag variable env exactly match test, you can configure the following collection policy:

```
accept tag.env == "test"
drop "*"
```

A collection policy consists of multi-line statements and contains special characters such as double quotation marks ("). If you configure a collection policy in Log Audit Service in the Log Service console, the system automatically escapes special characters in the policy. However, if you use Terraform to configure a collection policy, you must manually escape special characters and wrap lines. You can use one of the following methods to complete the configuration:

- Use EOF. For more information, see Configuration Syntax.

```
variable vpcflow_policy {
  type        = string
  default     = <<EOF
accept tag.env == \"test\"
drop \"*\"
EOF
}
resource "alicloud_log_audit" "example" {
  display_name = "tf-audit-test"
  aliuid       = "1234************"
  variable_map = {
    "vpc_flow_enabled" = "true",
    "vpc_flow_ttl" = "7",
    "vpc_sync_enabled" = "true",
    "vpc_sync_ttl" = "180"
    "vpc_flow_collection_policy" = var.vpcflow_policy
  }
  #if using rd custom mode for multi-account
  multi_account = ["1235************","1236************"]
  resource_directory_type="custom"
}
```

- Escape backslashes (\) and double quotation marks (") and wrap lines based on \n. For more information, see Built-in Functions.

```
resource "alicloud_log_audit" "example" {
  display_name = "tf-audit-test"
  aliuid       = "1234************"
  variable_map = {
    "vpc_flow_enabled" = "true",
    "vpc_flow_ttl" = "7",
    "vpc_sync_enabled" = "true",
    "vpc_sync_ttl" = "180"
    "vpc_flow_collection_policy" = "accept tag.env == \\\"test\\\"\\ndrop \\\"*\\\""
  }
  #if using rd custom mode for multi-account
  multi_account = ["1235************","1236************"]
  resource_directory_type="custom"
}
```

Appendix: parameters in variable_map

Parameter	Description	Default value

Parameter	Description	Default value
actiontrail_enabled	Specifies whether to collect ActionTrail logs. Valid values: • *true*: The system collects ActionTrail logs. • *false*: The system does not collect ActionTrail logs.	false
actiontrail_ttl	The retention period of ActionTrail logs in the central Logstore. Unit: days.	180
actiontrail_ti_enabled	Specifies whether to enable the threat intelligence feature for ActionTrail logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
oss_access_enabled	Specifies whether to collect Object Storage Service (OSS) access logs. Valid values: • *true*: The system collects OSS access logs. • *false*: The system does not collect OSS access logs.	false
oss_access_ttl	The retention period of OSS access logs in the regional Logstore. Unit: days.	7
oss_sync_enabled	Specifies whether to synchronize OSS access logs to the central project. Valid values: • *true*: The system synchronizes OSS access logs to the central project. • *false*: The system does not synchronize OSS access logs to the central project.	true
oss_sync_ttl	The retention period of OSS access logs in the central Logstore. Unit: days.	180
oss_access_ti_enabled	Specifies whether to enable the threat intelligence feature for OSS access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
oss_metering_enabled	Specifies whether to collect OSS metering logs. Valid values: • *true*: The system collects OSS metering logs. • *false*: The system does not collect OSS metering logs.	false

Parameter	Description	Default value
oss_metering_ttl	The retention period of OSS metering logs in the central Logstore. Unit: days.	180
rds_enabled	Specifies whether to collect ApsaraDB RDS for MySQL audit logs. Valid values: • *true*: The system collects ApsaraDB RDS for MySQL audit logs. • *false*: The system does not collect ApsaraDB RDS for MySQL audit logs.	false
rds_audit_collection_policy	The collection policy for ApsaraDB RDS for MySQL audit logs.	""
rds_ttl	The retention period of ApsaraDB RDS for MySQL audit logs in the central Logstore. Unit: days.	180
rds_ti_enabled	Specifies whether to enable the threat intelligence feature for ApsaraDB RDS for MySQL audit logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
rds_slow_enabled	Specifies whether to collect ApsaraDB RDS for MySQL slow query logs. Valid values: • *true*: The system collects ApsaraDB RDS for MySQL slow query logs. • *false*: The system does not collect ApsaraDB RDS for MySQL slow query logs.	false
rds_slow_collection_policy	The collection policy for ApsaraDB RDS for MySQL slow query logs.	""
rds_slow_ttl	The retention period of ApsaraDB RDS for MySQL slow query logs in the central Logstore. Unit: days.	180
rds_error_enabled	Specifies whether to collect ApsaraDB RDS for MySQL error logs. Valid values: • *true*: The system collects ApsaraDB RDS for MySQL error logs. • *false*: The system does not collect ApsaraDB RDS for MySQL error logs.	false
rds_error_collection_policy	The collection policy for ApsaraDB RDS for MySQL error logs.	""
rds_error_ttl	The retention period of ApsaraDB RDS for MySQL error logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
rds_perf_enabled	Specifies whether to collect ApsaraDB RDS for MySQL performance logs. Valid values: • *true*: The system collects ApsaraDB RDS for MySQL performance logs. • *false*: The system does not collect ApsaraDB RDS for MySQL performance logs.	false
rds_perf_collection_policy	The collection policy for ApsaraDB RDS for MySQL performance logs.	""
rds_perf_ttl	The retention period of ApsaraDB RDS for MySQL performance logs in the central Logstore. Unit: days.	180
vpc_flow_enabled	Specifies whether to collect VPC flow logs. Valid values: • *true*: The system collects VPC flow logs. • *false*: The system does not collect VPC flow logs.	false
vpc_flow_ttl	The retention period of VPC flow logs in the regional Logstore. Unit: days.	7
vpc_flow_collection_policy	The collection policy for VPC flow logs.	""
vpc_sync_enabled	Specifies whether to synchronize VPC flow logs to the central project. Valid values: • *true*: The system synchronizes VPC flow logs to the central project. • *false*: The system does not synchronize VPC flow logs to the central project.	true
vpc_sync_ttl	The retention period of VPC flow logs in the central Logstore. Unit: days.	180
polardb_enabled	Specifies whether to collect PolarDB for MySQL audit logs. Valid values: • *true*: The system collects PolarDB for MySQL audit logs. • *false*: The system does not collect PolarDB for MySQL audit logs.	false
polardb_audit_collection_policy	The collection policy for PolarDB for MySQL audit logs.	""
polardb_ttl	The retention period of PolarDB for MySQL audit logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
polardb_ti_enabled	Specifies whether to enable the threat intelligence feature for PolarDB for MySQL audit logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
polardb_slow_enabled	Specifies whether to collect PolarDB for MySQL slow query logs. Valid values: • *true*: The system collects PolarDB for MySQL slow query logs. • *false*: The system does not collect PolarDB for MySQL slow query logs.	false
polardb_slow_collection_policy	The collection policy for PolarDB for MySQL slow query logs.	""
polardb_slow_ttl	The retention period of PolarDB for MySQL slow query logs in the central Logstore. Unit: days.	180
polardb_error_enabled	Specifies whether to collect PolarDB for MySQL error logs. Valid values: • *true*: The system collects PolarDB for MySQL error logs. • *false*: The system does not collect PolarDB for MySQL error logs.	false
polardb_error_collection_policy	The collection policy for PolarDB for MySQL error logs.	""
polardb_error_ttl	The retention period of PolarDB for MySQL error logs in the central Logstore. Unit: days.	180
polardb_perf_enabled	Specifies whether to collect PolarDB for MySQL performance logs. Valid values: • *true*: The system collects PolarDB for MySQL performance logs. • *false*: The system does not collect PolarDB for MySQL performance logs.	false
polardb_perf_collection_policy	The collection policy for PolarDB for MySQL performance logs.	""
polardb_perf_ttl	The retention period of PolarDB for MySQL performance logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
drds_audit_enabled	Specifies whether to collect PolarDB-X 1.0 audit logs. Valid values: • *true*: The system collects PolarDB-X 1.0 audit logs. • *false*: The system does not collect PolarDB-X 1.0 audit logs.	false
drds_audit_collection_policy	The collection policy for PolarDB-X 1.0 audit logs.	""
drds_audit_ttl	The retention period of PolarDB-X 1.0 audit logs in the regional Logstore. Unit: days.	7
drds_sync_enabled	Specifies whether to synchronize PolarDB-X 1.0 audit logs to the central project. Valid values: • *true*: The system synchronizes PolarDB-X 1.0 audit logs to the central project. • *false*: The system does not synchronize PolarDB-X 1.0 audit logs to the central project.	true
drds_sync_ttl	The retention period of PolarDB-X 1.0 audit logs in the central Logstore. Unit: days.	180
slb_access_enabled	Specifies whether to collect Server Load Balancer (SLB) access logs. Valid values: • *true*: The system collects SLB access logs. • *false*: The system does not collect SLB access logs.	false
slb_access_collection_policy	The collection policy for SLB access logs.	""
slb_access_ttl	The retention period of SLB access logs in the regional Logstore. Unit: days.	7
slb_sync_enabled	Specifies whether to synchronize SLB access logs to the central project. Valid values: • *true*: The system synchronizes SLB access logs to the central project. • *false*: The system does not synchronize SLB access logs to the central project.	true
slb_sync_ttl	The retention period of SLB access logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
slb_access_ti_enabled	Specifies whether to enable the threat intelligence feature for SLB access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
alb_access_enabled	Specifies whether to collect Application Load Balancer (ALB) access logs. Valid values: • *true*: The system collects ALB access logs. • *false*: The system does not collect ALB access logs.	false
alb_access_collection_policy	The collection policy for ALB access logs.	""
alb_access_ttl	The retention period of ALB access logs in the regional Logstore. Unit: days.	7
alb_sync_enabled	Specifies whether to synchronize ALB access logs to the central project. Valid values: • *true*: The system synchronizes ALB access logs to the central project. • *false*: The system does not synchronize ALB access logs to the central project.	true
alb_sync_ttl	The retention period of ALB access logs in the central Logstore. Unit: days.	180
bastion_enabled	Specifies whether to collect Bastionhost operation logs. Valid values: • *true*: The system collects Bastionhost operation logs. • *false*: The system does not collect Bastionhost operation logs.	false
bastion_ttl	The retention period of Bastionhost operation logs in the central Logstore. Unit: days.	180
bastion_ti_enabled	Specifies whether to enable the threat intelligence feature for Bastionhost operation logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false

Parameter	Description	Default value
waf_enabled	Specifies whether to collect Web Application Firewall (WAF) access logs. Valid values: • *true*: The system collects WAF access logs. • *false*: The system does not collect WAF access logs.	false
waf_ttl	The retention period of WAF access logs in the central Logstore. Unit: days.	180
waf_ti_enabled	Specifies whether to enable the threat intelligence feature for WAF access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
cloudfirewall_enabled	Specifies whether to collect Internet firewall traffic logs for Cloud Firewall. Valid values: • *true*: The system collects Internet firewall traffic logs for Cloud Firewall. • *false*: The system does not collect Internet firewall traffic logs for Cloud Firewall.	false
cloudfirewall_ttl	The retention period of Cloud Firewall Internet firewall traffic logs in the central Logstore. Unit: days.	180
cloudfirewall_ti_enabled	Specifies whether to enable the threat intelligence feature for Cloud Firewall Internet firewall traffic logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
cloudfirewall_vpc_enabled	Specifies whether to collect VPC firewall traffic logs for Cloud Firewall. Valid values: • *true*: The system collects VPC firewall traffic logs for Cloud Firewall. • *false*: The system does not collect VPC firewall traffic logs for Cloud Firewall.	false
cloudfirewall_vpc_ttl	The retention period of Cloud Firewall VPC firewall traffic logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
cloudfirewall_vpc_ti_enabled	Specifies whether to enable the threat intelligence feature for Cloud Firewall VPC firewall traffic logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
ddos_coo_access_enabled	Specifies whether to collect Anti-DDoS Pro access logs. Valid values: • *true*: The system collects Anti-DDoS Pro access logs. • *false*: The system does not collect Anti-DDoS Pro access logs.	false
ddos_coo_access_ttl	The retention period of Anti-DDoS Pro access logs in the central Logstore. Unit: days.	180
ddos_coo_access_ti_enabled	Specifies whether to enable the threat intelligence feature for Anti-DDoS Pro access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
ddos_bgp_access_enabled	Specifies whether to collect Anti-DDoS Origin access logs. Valid values: • *true*: The system collects Anti-DDoS Origin access logs. • *false*: The system does not collect Anti-DDoS Origin access logs.	false
ddos_bgp_access_ttl	The retention period of Anti-DDoS Origin access logs in the central Logstore. Unit: days.	180
ddos_dip_access_enabled	Specifies whether to collect Anti-DDoS Premium access logs. Valid values: • *true*: The system collects Anti-DDoS Premium access logs. • *false*: The system does not collect Anti-DDoS Premium access logs.	false
ddos_dip_access_ttl	The retention period of Anti-DDoS Premium access logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
ddos_dip_access_ti_enabled	Specifies whether to enable the threat intelligence feature for Anti-DDoS Premium access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
sas_ttl	The retention period of Security Center (SAS) logs in the central Logstore. Unit: days.	180
sas_process_enabled	Specifies whether to collect SAS process startup logs. Valid values: • *true*: The system collects SAS process startup logs. • *false*: The system does not collect SAS process startup logs.	false
sas_network_enabled	Specifies whether to collect SAS network connection logs. Valid values: • *true*: The system collects SAS network connection logs. • *false*: The system does not collect SAS network connection logs.	false
sas_login_enabled	Specifies whether to collect SAS logon logs. Valid values: • *true*: The system collects SAS logon logs. • *false*: The system does not collect SAS logon logs.	false
sas_crack_enabled	Specifies whether to collect SAS brute-force attack logs. Valid values: • *true*: The system collects SAS brute-force attack logs. • *false*: The system does not collect SAS brute-force attack logs.	false
sas_snapshot_process_enabled	Specifies whether to collect SAS process snapshot logs. Valid values: • *true*: The system collects SAS process snapshot logs. • *false*: The system does not collect SAS process snapshot logs.	false

Parameter	Description	Default value
sas_snapshot_account_enabled	Specifies whether to collect SAS account snapshot logs. Valid values: • *true*: The system collects SAS account snapshot logs. • *false*: The system does not collect SAS account snapshot logs.	false
sas_snapshot_port_enabled	Specifies whether to collect SAS port snapshot logs. Valid values: • *true*: The system collects SAS port snapshot logs. • *false*: The system does not collect SAS port snapshot logs.	false
sas_dns_enabled	Specifies whether to collect SAS DNS logs. Valid values: • *true*: The system collects SAS DNS logs. • *false*: The system does not collect SAS DNS logs.	false
sas_local_dns_enabled	Specifies whether to collect SAS local DNS logs. Valid values: • *true*: The system collects SAS local DNS logs. • *false*: The system does not collect SAS local DNS logs.	false
sas_session_enabled	Specifies whether to collect SAS network session logs. Valid values: • *true*: The system collects SAS network session logs. • *false*: The system does not collect SAS network session logs.	false
sas_http_enabled	Specifies whether to collect SAS web access logs. Valid values: • *true*: The system collects SAS web access logs. • *false*: The system does not collect SAS web access logs.	false
sas_security_vul_enabled	Specifies whether to collect SAS vulnerability logs. Valid values: • *true*: The system collects SAS vulnerability logs. • *false*: The system does not collect SAS vulnerability logs.	false

Parameter	Description	Default value
sas_security_hc_enabled	Specifies whether to collect SAS baseline logs. Valid values: • *true*: The system collects SAS baseline logs. • *false*: The system does not collect SAS baseline logs.	false
sas_security_alert_enabled	Specifies whether to collect SAS security alert logs. Valid values: • *true*: The system collects SAS security alert logs. • *false*: The system does not collect SAS security alert logs.	false
sas_ti_enabled	Specifies whether to enable the threat intelligence feature for SAS logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
apigateway_enabled	Specifies whether to collect API Gateway access logs. Valid values: • *true*: The system collects API Gateway access logs. • *false*: The system does not collect API Gateway access logs.	false
apigateway_ttl	The retention period of API Gateway access logs in the central Logstore. Unit: days.	180
apigateway_ti_enabled	Specifies whether to enable the threat intelligence feature for API Gateway access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
nas_enabled	Specifies whether to collect Apsara File Storage NAS access logs. Valid values: • *true*: The system collects NAS access logs. • *false*: The system does not collect NAS access logs.	false
nas_ttl	The retention period of NAS access logs in the central Logstore. Unit: days.	180

Parameter	Description	Default value
nas_ti_enabled	Specifies whether to enable the threat intelligence feature for NAS access logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
appconnect_enabled	Specifies whether to collect Cloud Service Bus (CSB) App Connect logs. Valid values: • *true*: The system collects App Connect logs. • *false*: The system does not collect App Connect logs.	false
appconnect_ttl	The retention period of App Connect logs in the central Logstore. Unit: days.	180
appconnect_ti_enabled	Specifies whether to enable the threat intelligence feature for App Connect logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
cps_enabled	Specifies whether to collect Alibaba Cloud Mobile Push logs. Valid values: • *true*: The system collects Alibaba Cloud Mobile Push logs. • *false*: The system does not collect Alibaba Cloud Mobile Push logs.	false
cps_ttl	The retention period of Alibaba Cloud Mobile Push logs in the central Logstore. Unit: days.	180
cps_ti_enabled	Specifies whether to enable the threat intelligence feature for Alibaba Cloud Mobile Push logs. Valid values: • *true*: The system enables the threat intelligence feature. • *false*: The system disables the threat intelligence feature.	false
k8s_audit_enabled	Specifies whether to collect Kubernetes audit logs. Valid values: • *true*: The system collects Kubernetes audit logs. • *false*: The system does not collect Kubernetes audit logs.	false

Parameter	Description	Default value
k8s_audit_collection_policy	The collection policy for Kubernetes audit logs.	""
k8s_audit_ttl	The retention period of Kubernetes audit logs in the central Logstore. Unit: days.	180
k8s_event_enabled	Specifies whether to collect Kubernetes event logs. Valid values: • *true*: The system collects Kubernetes event logs. • *false*: The system does not collect Kubernetes event logs.	false
k8s_event_collection_policy	The collection policy for Kubernetes event logs.	""
k8s_event_ttl	The retention period of Kubernetes event logs in the central Logstore. Unit: days.	180
k8s_ingress_enabled	Specifies whether to collect Kubernetes Ingress access logs. Valid values: • *true*: The system collects Kubernetes Ingress access logs. • *false*: The system does not collect Kubernetes Ingress access logs.	false
k8s_ingress_collection_policy	The collection policy for Kubernetes Ingress access logs.	""
k8s_ingress_ttl	The retention period of Kubernetes Ingress access logs in the central Logstore. Unit: days.	180

1.10. Use a custom policy to authorize Log Service to collect and synchronize logs

The Log Audit Service application allows you to collect logs from Alibaba Cloud services across multiple Alibaba Cloud accounts. Before you can collect logs, you must authorize Log Service and the related accounts. To authorize Log Service, you can use the AccessKey pair of a RAM user who has the required permissions. You can also follow the steps described in this topic to create a custom policy in Resource Access Management (RAM).

Context

You can use the Log Audit Service application to collect cloud service logs of an Alibaba Cloud account or across multiple Alibaba Cloud accounts. To collect the logs of cloud services across multiple Alibaba Cloud accounts, you must grant mutual access between the current Alibaba Cloud account and the other Alibaba Cloud accounts.

> **Note** When the AliyunServiceRoleForSLSAudit service-linked role is created, the current Alibaba Cloud account is automatically authorized. For more information, see Initially configure Log Audit Service. If you want to authorize other Alibaba Cloud accounts by using a custom policy, you can perform the steps described in this topic.

- You must authorize the current Alibaba Cloud account to receive logs from other Alibaba Cloud accounts. The logs are stored in the Logstore that is dedicated to audit logs.
- You must authorize other Alibaba Cloud accounts to synchronize logs to the current Alibaba Cloud account. The logs are stored in the Logstore that is dedicated to audit logs.

The Log Audit Service application of Log Service involves multiple roles and policies. The following tables describes the relationships among the roles and policies.

- Current Alibaba Cloud account

Role	Policy
AliyunServiceRoleForSLSAudit	AliyunServiceRolePolicyForSLSAudit

- Other Alibaba Cloud accounts

Role	Policy
sls-audit-service-monitor	ReadOnlyAccess AliyunLogAuditServiceMonitorAccess

Procedure

1. Use one of the other Alibaba Cloud accounts to log on to the RAM console.

   We recommend that you use a RAM user to complete authorization. The RAM user must be granted the read and write permissions on RAM resources. To grant the required permissions to the RAM user, you can attach the *AliyunRAMFullAccess* policy to the RAM user.

2. Create a policy named AliyunLogAuditServiceMonitorAccess.
   i. In the left-side navigation pane, choose **Permissions > Policies**. On the page that appears, click **Create Policy**.

ii. On the **Create Custom Policy** page, set the parameters and click **OK**. The following table describes the parameters.

Parameter	Description
Policy Name	Set the value to **AliyunLogAuditServiceMonitorAccess**.
Configuration Mode	Select **Script**.
Policy Document	The content of the policy. Replace the content in the editor with the following script: `{ "Version": "1", "Statement": [{ "Action": "log:*", "Resource": ["acs:log:*:*:project/slsaudit-*", "acs:log:*:*:app/audit"], "Effect": "Allow" }, { "Action": ["rds:ModifySQLCollectorPolicy", "vpc:*FlowLog*", "drds:*SqlAudit*", "kvstore:ModifyAuditLogConfig", "polardb:ModifyDBClusterAuditLogCollector"], "Resource": "*", "Effect": "Allow" }] }`

3. Create a role named *sls-audit-service-monitor*.
   i. In the left-side navigation pane, choose **Identities > Roles**. On the page that appears, click **Create Role**.
   ii. In the **Select Role Type** step, select **Alibaba Cloud Service** and click **Next**.

iii. In the **Configure Role** step, set the parameters and click **OK**. The following table describes the parameters.

Parameter	Description
Role Type	Select **Normal Service Role**.
RAM Role Name	Set the value to **sls-audit-service-monitor**.
Select Trusted Service	Select **Log Service** from the drop-down list.

iv. In the **Finish** step, click **Add Permissions to RAM Role**.

4. Attach the AliyunLogAuditServiceMonitorAccess policy to the sls-audit-service-monitor role.

   In the **Add Permissions** panel, click **Custom Policy** in the Select Policy section and select the **AliyunLogAuditServiceMonitorAccess** policy. Then, click **System Policy** and select the **ReadOnlyAccess** policy. Click **OK**.

5. Modify the trust policy of the sls-audit-service-monitor role.

   i. On the Roles page, find and click the **sls-audit-service-monitor** role to go to the details page of the role.

   ii. Click the **Trust Policy Management** tab. On the tab, replace the content in the editor with the following script and click **OK**.

   Replace Alibaba Cloud account ID with the actual ID. You can view the ID of your Alibaba Cloud account in the Account Management console.

```
{
   "Statement": [
       {
           "Action": "sts:AssumeRole",
           "Effect": "Allow",
           "Principal": {
               "Service": [
                   "Alibaba Cloud account ID@log.aliyuncs.com",
                   "log.aliyuncs.com"
               ]
           }
       }
   ],
   "Version": "1"
}
```

1.11. Log fields

1.11.1. ActionTrail

This topic describes the fields of operation logs in ActionTrail.

Log field	Description
__topic__	The topic of a log entry. Valid value: actiontrail_event.

Log field	Description
owner_id	The ID of an Alibaba Cloud account.
event	The log event in the JSON format. The content of this field varies based on the log event.
event.eventId	The ID of an event.
event.eventName	The name of an event.
event.eventSource	The source of an event.
event.eventType	The type of an event.
event.eventVersion	The data format version of an event. Valid value: 1.
event.acsRegion	The region where an event occurs.
event.requestId	The ID of an API request.
event.apiVersion	The version of the API.
event.errorMessage	The error message of an event.
event.serviceName	The name of the Alibaba Cloud service that is associated with an event.
event.sourceIpAddress	The source IP address that is associated with an event.
event.userAgent	The User-Agent HTTP header that is associated with an event.
event.requestParameters.HostId	The ID of the host from which a request is sent.
event.requestParameters.Name	The name of a request parameter.
event.requestParameters.Region	The region from which a request is sent.
event.userIdentity.accessKeyId	The AccessKey ID of an account that sends a request.
event.userIdentity.accountId	The ID of an account that sends a request.
event.userIdentity.principalId	The principal ID of an account that sends a request.
event.userIdentity.type	The type of an account that sends a request.
event.userIdentity.userName	The username of an account that sends a request.
event.errorCode	The error code of an event.
addionalEventData.isMFAChecked	Indicates whether multi-factor authentication (MFA) is enabled for the account that is used to log on to Log Service.
addionalEventData.loginAccount	The logon account.

1.11.2. OSS

This topic describes the fields in the log entries that are generated by Object Storage Service (OSS).

- Access logs

  Access logs record all incoming traffic to each OSS bucket. Access log entries are collected in real time. The following table describes the fields in an access log entry.

Field	Description
__topic__	The topic of the log entry. Valid value: oss_access_log.
owner_id	The ID of the Alibaba Cloud account that owns the bucket.
region	The region where the bucket resides.
access_id	The AccessKey ID of the requester.
time	The time at which OSS receives the request. If you want to record timestamps, use the _time_ field.
owner_id	The ID of the Alibaba Cloud account that owns the bucket.
User-Agent	The User-Agent field in the header of the request.
logging_flag	Indicates whether logging is enabled to export logs to a bucket at regular intervals.
bucket	The name of the bucket.
content_length_in	The value of the Content-Length field in the header of the request. Unit: bytes.
content_length_out	The value of the Content-Length field in the header of the response. Unit: bytes.
object	The URL-encoded object that is requested. You can include the select url_decode(object) clause in your query statement to decode the object.
object_size	The size of the requested object. Unit: bytes.
operation	The type of the operation that is specified in the request. For more information, see the "Operation types" section of this topic.
request_uri	The URL-encoded URI of the request. The value of this field includes a query string. You can include the select url_decode(request_uri) clause in your query statement to decode the URI.
error_code	The error code that is returned by OSS. For more information, see Error responses.
request_length	The size of the request. The size includes the header of the request. Unit: bytes.

Field	Description
client_ip	The IP address from which the request is sent. The IP address can be the IP address of a client, firewall, or proxy.
response_body_length	The size of the body of the response. The size excludes the header of the response.
http_method	The method that is used to send the request.
referer	The Referer field in the header of the request.
requester_id	The ID of the Alibaba Cloud account that sends the request. If you use an anonymous logon, the value of this field is a hyphen (-).
request_id	The ID of the request.
response_time	The amount of time that is taken by OSS to respond to the request. Unit: milliseconds.
server_cost_time	The amount of time that is taken by OSS to process the request. Unit: milliseconds.
http_type	The protocol over which the request is sent. Valid values: HTTP and HTTPS.
sign_type	The type of the signature in the request. For more information, see the "Signature types" section of this topic.
http_status	The status code in the response that is returned by OSS.
sync_request	The type of synchronous communication for the request. For more information, see the "Synchronous request types" section of this topic.
bucket_storage_type	The storage class of the OSS object. For more information, see storage classes.
host	The domain name that is requested.
vpc_addr	The virtual private cloud (VPC) IP address that maps the requested domain name.
vpc_id	The ID of the VPC to which the bucket belongs.
delta_data_size	The amount of change in the size of the requested object. If the size of the object does not change, the value of this field is 0. If the request does not specify an upload operation, the value of this field is a hyphen (-).
acc_access_region	If the request specifies to accelerate data transmission, the value of this field is the ID of the region where the requested access point resides. Otherwise, the value of this field is a hyphen (-).

Field	Description
restore_priority	The priority of the log entry in the event of data restoration.

- Batch deletion logs

  Batch deletion logs record the operations that are performed to delete multiple objects at a time from each bucket. Batch deletion log entries are collected in real time. The following table describes the fields in a batch deletion log entry.

  > **Note** When you call the DeleteObjects operation, an access log entry is generated to record the operation. The information about the objects that you specify in the DeleteObjects operation is carried in the body of the request. Therefore, the value of the object field in the generated access log entry is a hyphen (-). If you want to view the information about the objects that you specify in the DeleteObjects operation, you can query the batch deletion log entries in which the value of the request_id field matches the ID of the request.

Field	Description
__topic__	The topic of the log entry. Valid value: oss_batch_delete_log.
owner_id	The ID of the Alibaba Cloud account that owns the bucket.
region	The region where the bucket resides.
client_ip	The IP address from which the request is sent. The IP address can be the IP address of a client, firewall, or proxy.
user_agent	The User-Agent field in the header of the request.
bucket	The name of the bucket.
error_code	The error code that is returned by OSS. For more information, see the "Synchronous request types" section of this topic.
request_length	The size of the body of the request. The size includes the header of the request. Unit: bytes.
response_body_length	The size of the body of the response. The size excludes the header of the response.
object	The URL-encoded object that is requested. You can include the select url_decode(object) clause in your query statement to decode the object.
object_size	The size of the requested object. Unit: bytes.
operation	The type of the operation that is specified in the request. For more information, see the "Operation types" section of this topic.
bucket_location	The cluster to which the bucket belongs.
http_method	The method that is used to send the request.

Field	Description
referer	The Referer field in the header of the request.
request_id	The ID of the request.
http_status	The status code in the response that is returned by OSS.
sync_request	The type of synchronous communication for the request. For more information, see the "Synchronous request types" section of this topic.
request_uri	The URL-encoded URI of the request. The value of this field includes a query string. You can include the select url_decode(request_uri) clause in your query statement to decode the URI.
host	The domain name that is requested.
logging_flag	Indicates whether logging is enabled to export logs to the bucket at regular intervals.
server_cost_time	The amount of time that is taken by OSS to process the request. Unit: milliseconds.
owner_id	The ID of the Alibaba Cloud account that owns the bucket.
requester_id	The ID of the Alibaba Cloud account that sends the request. If you use an anonymous logon, the value of this field is a hyphen (-).
delta_data_size	The amount of change in the size of the requested object. If the size of the object does not change, the value of this field is 0. If the request does not specify an upload operation, the value of this field is a hyphen (-).

- Hourly metering logs

  Hourly metering logs record the hourly metering operations on each bucket. Hourly metering log entries are collected at a latency of a few hours. The following table describes the fields in an hourly metering log entry.

Field	Description
__topic__	The topic of the log entry. Valid value: oss_metering_log.
owner_id	The ID of the Alibaba Cloud account that owns the bucket.
bucket	The name of the bucket.
cdn_in	The volume of inbound traffic from CDN to the bucket. Unit: bytes.
cdn_out	The volume of outbound traffic from the bucket to CDN. Unit: bytes.
get_request	The number of GET requests that are sent to request resources from the bucket.

Field	Description
intranet_in	The volume of inbound traffic to the bucket over internal networks. Unit: bytes.
intranet_out	The volume of outbound traffic from the bucket over internal networks. Unit: bytes.
network_in	The volume of inbound traffic to the bucket over the Internet. Unit: bytes.
network_out	The volume of outbound traffic from the bucket over the Internet. Unit: bytes.
put_request	The number of PUT requests that are sent to request resources from the bucket.
storage_type	The storage class of the OSS bucket. For more information, see the "storage classes" section of this topic.
storage	The amount of data in the bucket. Unit: bytes.
metering_datasize	The amount of data of a storage class rather than the Standard storage class in the bucket.
process_img_size	The size of the image that is processed. Unit: bytes.
process_img	The image that is processed.
sync_in	The volume of inbound traffic to the bucket over synchronous requests. Unit: bytes.
sync_out	The volume of outbound traffic from the bucket over synchronous requests. Unit: bytes.
start_time	The time at which the metering operation starts.
end_time	The time at which the metering operation ends.
region	The region where the bucket resides.

storage classes

Storage class	Description
standard	The Standard storage class.
archive	The Archive storage class.
infrequent_access	The Infrequent Access (IA) storage class.

For more information about the related API operations, see List of operations by function.

Operation types

Operation	Description
AbortMultiPartUpload	Aborts a multipart upload task.
AppendObject	Appends an object.
CompleteUploadPart	Completes a multipart upload task.
CopyObject	Copies an object.
DeleteBucket	Deletes a bucket.
DeleteLiveChannel	Deletes a live channel.
DeleteObject	Deletes an object.
DeleteObjects	Deletes multiple objects.
GetBucket	Queries all objects in a bucket.
GetBucketAcl	Queries the access control list (ACL) of a bucket.
GetBucketCors	Queries the cross-origin resource sharing (CORS) rules of a bucket.
GetBucketEventNotification	Queries the notification configurations of a bucket.
GetBucketInfo	Queries the information about a bucket.
GetBucketLifecycle	Queries the lifecycle rules of a bucket.
GetBucketLocation	Queries the region where a bucket resides.
GetBucketLog	Queries the access log configurations of a bucket.
GetBucketReferer	Queries the hotlink protection configurations of a bucket.
GetBucketReplication	Queries the cross-region replication (CRR) rules of a bucket.
GetBucketReplicationProgress	Queries the progress of a CRR task that is performed on a bucket.
GetBucketStat	Queries the status of a bucket.
GetBucketWebSite	Queries the status of the static website hosting feature for a bucket.
GetLiveChannelStat	Queries the status of a live channel.
GetObject	Reads an object.
GetObjectAcl	Queries the ACL of an object.
GetObjectInfo	Queries the information about an object.
GetObjectMeta	Queries the metadata of an object.

Operation	Description
GetObjectSymlink	Queries the symbolic link of an object.
GetPartData	Queries the data in all parts of an object.
GetPartInfo	Queries the information about all parts of an object.
GetProcessConfiguration	Queries the image processing configurations of a bucket.
GetService	Queries all buckets that are created within your Alibaba Cloud account.
HeadBucket	Queries the information about a bucket.
HeadObject	Queries the metadata of an object.
InitiateMultipartUpload	Initializes a multipart upload task.
ListMultiPartUploads	Queries all multipart upload events of a bucket.
ListParts	Queries the status of all parts of an object.
PostObject	Uploads an object by using a form.
PostProcessTask	Commits a task to process data, such as screenshots.
PostVodPlaylist	Creates a video-on-demand (VOD) playlist for a live channel.
ProcessImage	Processes an image.
PutBucket	Creates a bucket.
PutBucketCors	Configures a CORS rule for a bucket.
PutBucketLifecycle	Configures the lifecycle of a bucket.
PutBucketLog	Configures an access log for a bucket.
PutBucketWebSite	Configures the static website hosting feature for a bucket.
PutLiveChannel	Creates a live channel.
PutLiveChannelStatus	Specifies the status of a live channel.
PutObject	Uploads an object.
PutObjectAcl	Modifies the ACL of an object.
PutObjectSymlink	Creates a symbolic link for an object.
RedirectBucket	Redirects a request to the endpoint of a different bucket.
RestoreObject	Restores an object.

Operation	Description
UploadPart	Resumes a multipart upload task from a specified checkpoint.
UploadPartCopy	Copies a part of an object.
get_image_exif	Queries the exchangeable image file format (EXIF) data of an image.
get_image_info	Queries the length and width of an image.
get_image_infoexif	Queries the length, width, and Exif data of an image.
get_style	Queries the bucket style of a live channel.
list_style	Queries all bucket styles that are available for a live channel.
put_style	Creates a bucket style.

Synchronous request types

Synchronous request type	Description
Hyphen (-)	The request is a general request.
cdn	The request is a CDN back-to-origin request.

For more information about signatures, see Verify user signatures.

Signature types

Signature type	Description
NotSign	The request is unsigned.
NormalSign	The request is signed with a regular signature.
UriSign	The request is signed with a URL signature.
AdminSign	The request is signed with an administrator account.

1.11.3. ApsaraDB RDS

This topic describes the fields of audit logs, slow query logs, error logs, and performance logs in ApsaraDB RDS.

Audit logs

Field	Description
__topic__	The topic of the log. The value is fixed as rds_audit_log.

Field	Description
owner_id	The ID of the Alibaba Cloud account to which the ApsaraDB RDS instance belongs.
region	The ID of the region where the ApsaraDB RDS instance resides.
instance_name	The name of the ApsaraDB RDS instance.
instance_id	The ID of the ApsaraDB RDS instance.
db_type	The type of the databases that are created on the ApsaraDB RDS instance.
db_version	The version of the database engine.
check_rows	The number of scanned rows.
db	The name of the database.
fail	Indicates whether the SQL statement is successfully executed. Valid values: • 0: successful • 1: failed
client_ip	The IP address of the client that accesses the ApsaraDB RDS instance.
latency	The time that is consumed to return the result of the SQL statement. Unit: microseconds.
origin_time	The point in time at which the SQL statement is executed.
return_rows	The number of returned rows.
sql	The SQL statement that is executed.
thread_id	The ID of the thread.
user	The name of the user who executes the SQL statement.
update_rows	The number of updated rows.

Slow query logs

Field	Description
__topic__	The topic of the log. The value is fixed as rds_slow_log.
owner_id	The ID of the Alibaba Cloud account to which the ApsaraDB RDS instance belongs.
region	The ID of the region where the ApsaraDB RDS instance resides.

Field	Description
instance_name	The name of the ApsaraDB RDS instance.
instance_id	The ID of the ApsaraDB RDS instance.
db_type	The type of the databases that are created on the ApsaraDB RDS instance.
db_version	The version of the database engine.
db_name	The name of the database.
rows_examined	The number of scanned rows.
rows_sent	The number of returned rows.
start_time	The point in time at which the SQL statement is executed.
query_time	The time that is consumed to execute the SQL statement. Unit: seconds.
lock_time	The duration of the lock wait. Unit: seconds.
user_host	The information about the client.
query_sql	The SQL statement of the slow query.

Performance logs

Metric	Description
mysql_perf_active_session	The number of active connections.
mysql_perf_com_delete	The average number of times that DELETE statements are executed per second.
mysql_perf_com_insert	The average number of times that INSERT statements are executed per second.
mysql_perf_com_insert_select	The average number of times that INSERT SELECT statements are executed per second.
mysql_perf_com_replace	The average number of times that REPLACE statements are executed per second.
mysql_perf_com_replace_select	The average number of times that REPLACE SELECT statements are executed per second.
mysql_perf_com_select	The average number of times that SELECT statements are executed per second.
mysql_perf_com_update	The average number of times that UPDATE statements are executed per second.

Metric	Description
mysql_perf_conn_usage	The connection utilization of the ApsaraDB RDS instance. Unit: percent.
mysql_perf_cpu_usage	The CPU utilization of the ApsaraDB RDS instance. Unit: percent.
mysql_perf_data_size	The amount of data that is used by the ApsaraDB RDS instance. Unit: MB.
mysql_perf_disk_usage	The disk utilization of the ApsaraDB RDS instance. Unit: percent.
mysql_perf_ibuf_dirty_ratio	The ratio of dirty pages to the total number of pages in the buffer pool. Unit: percent.
mysql_perf_ibuf_read_hit	The read hit ratio of the buffer pool. Unit: percent.
mysql_perf_ibuf_request_r	The average number of read operations that are performed on the InnoDB buffer pool per second.
mysql_perf_ibuf_request_w	The average number of write operations that are performed on the InnoDB buffer pool per second.
mysql_perf_ibuf_use_ratio	The utilization of the buffer pool. Unit: percent.
mysql_perf_inno_data_read	The amount of data that InnoDB reads per second. Unit: KB.
mysql_perf_inno_data_written	The amount of data that InnoDB writes per second. Unit: KB.
mysql_perf_inno_row_delete	The average number of rows that are deleted from the InnoDB table per second.
mysql_perf_inno_row_insert	The average number of rows that are inserted from the InnoDB table per second.
mysql_perf_inno_row_readed	The average number of rows that are read from the InnoDB table per second.
mysql_perf_inno_row_update	The average number of rows that are updated from the InnoDB table per second.
mysql_perf_innodb_log_write_requests	The average number of log write requests per second.
mysql_perf_innodb_log_writes	The average number of physical writes to the log file per second.
mysql_perf_innodb_os_log_fsyncs	The average number of write operations to the log file by calling the fsync() function.
mysql_perf_ins_size	The disk space that is occupied by the ApsaraDB RDS instance. Unit: MB.
mysql_perf_iops	The input/output operations per second (IOPS).

Metric	Description
mysql_perf_iops_usage	The IOPS utilization of the ApsaraDB RDS instance. Unit: percent.
mysql_perf_kbytes_received	The average inbound traffic per second. Unit: KB.
mysql_perf_kbytes_sent	The average outbound traffic per second. Unit: KB.
mysql_perf_log_size	The amount of binary logs that are generated by the ApsaraDB RDS instance. Unit: MB.
mysql_perf_mem_usage	The memory usage of the ApsaraDB RDS instance. Unit: percent.
mysql_perf_open_tables	The number of opened tables.
mysql_perf_other_size	The amount of other resources that are used by the ApsaraDB RDS instance. Unit: MB.
mysql_perf_qps	The average number of times that SQL statements are executed per second.
mysql_perf_slow_queries	The average number of slow queries per second.
mysql_perf_tb_tmp_disk	The number of temporary tables that are automatically created per second in the disk when SQL statements are executed in the MySQL process.
mysql_perf_threads_connected	The number of open MySQL connections.
mysql_perf_threads_running	The number of MySQL threads that are not sleeping.
mysql_perf_tmp_size	The amount of storage space that is occupied by the temporary files of the ApsaraDB RDS instance. Unit: MB.
mysql_perf_total_session	The total number of connections.
mysql_perf_tps	The average number of transactions per second.

Error logs

Field	Description
__topic__	The topic of the log. The value is fixed as rds_error_log.
collect_time	The time at which the log is collected.
content	The content of the log.
db_type	The type of the databases that are created on the ApsaraDB RDS instance.
db_version	The version of the database engine.

Field	Description
event_type	The type of the event.
instance_id	The ID of the ApsaraDB RDS instance.
instance_name	The name of the ApsaraDB RDS instance.
region	The ID of the region where the ApsaraDB RDS instance resides.

1.11.4. PolarDB for MySQL

This topic describes the fields of audit logs, slow query logs, error logs, and performance logs in PolarDB for MySQL.

Audit logs

Field	Description
__topic__	The topic of the log. The value is fixed as polardb_audit_log.
owner_id	The ID of the Alibaba Cloud account to which the PolarDB for MySQL cluster belongs.
region	The ID of the region where the PolarDB for MySQL cluster resides.
cluster_id	The ID of the PolarDB for MySQL cluster.
node_id	The ID of the node in the PolarDB for MySQL cluster.
check_rows	The number of scanned rows.
db	The name of the database.
fail	Indicates whether the SQL statement is successfully executed. Valid values: • 0: successful • 1: failed
client_ip	The IP address of the client that accesses the PolarDB for MySQL cluster.
latency	The time that is consumed to return the result of the SQL statement. Unit: microseconds.
origin_time	The point in time at which the SQL statement is executed.
return_rows	The number of returned rows.
sql	The SQL statement that is executed.
thread_id	The ID of the thread.

Field	Description
user	The name of the user who executes the SQL statement.
update_rows	The number of updated rows.

Slow query logs

Field	Description
__topic__	The topic of the log. The value is fixed as polardb_slow_log.
owner_id	The ID of the Alibaba Cloud account to which the PolarDB for MySQL cluster belongs.
region	The ID of the region where the PolarDB for MySQL cluster resides.
cluster_id	The ID of the PolarDB for MySQL cluster.
node_id	The ID of the node in the PolarDB for MySQL cluster.
db_type	The type of the databases that are created in the PolarDB for MySQL cluster.
db_name	The name of the PolarDB for MySQL database.
version	The version of the database engine.
rows_examined	The number of scanned rows.
rows_sent	The number of returned rows.
start_time	The point in time at which the SQL statement is executed.
query_time	The time that is consumed to execute the SQL statement. Unit: seconds.
lock_time	The duration of the lock wait. Unit: seconds.
user_host	The information about the client.
query_sql	The SQL statement of the slow query.

Error logs

Field	Description
__topic__	The topic of the log. The value is fixed as polardb_error_log.
collect_time	The time at which the log is collected.
content	The content of the log.

Field	Description
type_role	The type of the PolarDB for MySQL cluster.
event_type	The type of the event.
instance_id	The ID of the PolarDB for MySQL cluster.
region	The ID of the region where the PolarDB for MySQL cluster resides.
db_version	The version of the database engine.

Performance logs

Metric	Description
mysql_perf_active_session	The number of active connections per second.
mysql_perf_binlog_size	The amount of binary logs that are generated on your computer. Unit: MB.
mysql_perf_com_delete	The average number of times that DELETE statements are executed per second.
mysql_perf_com_delete_multi	The average number of times that MULTI-DELETE statements are executed per second.
mysql_perf_com_insert	The average number of times that INSERT statements are executed per second.
mysql_perf_com_insert_select	The average number of times that INSERT-SELECT statements are executed per second.
mysql_perf_com_replace	The average number of times that REPLACE statements are executed per second.
mysql_perf_com_replace_select	The average number of times that REPLACE-SELECT statements are executed per second.
mysql_perf_com_select	The average number of times that SELECT statements are executed per second.
mysql_perf_com_update	The average number of times that UPDATE statements are executed per second.
mysql_perf_com_update_multi	The average number of times that MULTI-UPDATE statements are executed per second.
mysql_perf_cpu_ratio	The CPU utilization. Unit: percent.
mysql_perf_created_tmp_disk_tables	The number of temporary tables that are automatically created per second.
mysql_perf_data_size	The amount of storage space that is occupied by data. Unit: MB.

Metric	Description
mysql_perf_innodb_buffer_dirty_ratio	The ratio of dirty pages to the total number of pages in the buffer pool. Unit: percent.
mysql_perf_innodb_buffer_read_hit	The read hit ratio of the buffer pool. Unit: percent.
mysql_perf_innodb_buffer_use_ratio	The utilization of the buffer pool. Unit: percent.
mysql_perf_innodb_data_read	The amount of data that is read from the storage engine per second. Unit: bytes.
mysql_perf_innodb_data_reads	The average number of read operations that are performed on the buffer pool per second.
mysql_perf_innodb_data_writes	The average number of write operations that are performed on the buffer pool per second.
mysql_perf_innodb_data_written	The amount of data that is written to the storage engine per second. Unit: bytes.
mysql_perf_innodb_log_write_requests	The average number of log write requests per second.
mysql_perf_innodb_os_log_fsyncs	The average number of write operations to the log file by calling the fsync() function.
mysql_perf_innodb_rows_deleted	The number of deleted rows per second.
mysql_perf_innodb_rows_inserted	The number of inserted rows per second.
mysql_perf_innodb_rows_read	The number of read rows per second.
mysql_perf_iops	The input/output operations per second (IOPS).
mysql_perf_iops_r	The read IOPS.
mysql_perf_iops_throughput	The I/O throughput per second. Unit: MB.
mysql_perf_iops_throughput_r	The read I/O throughput per second. Unit: MB.
mysql_perf_iops_throughput_w	The write I/O throughput per second. Unit: MB.
mysql_perf_iops_w	The write IOPS.
mysql_perf_kbytes_received	The average inbound traffic per second. Unit: KB.
mysql_perf_kbytes_sent	The average outbound traffic per second. Unit: KB.
mysql_perf_mem_ratio	The memory usage. Unit: percent.

Metric	Description
mysql_perf_mps	The number of operations that are performed on data per second.
mysql_perf_other_log_size	The amount of other logs that are generated. Unit: MB.
mysql_perf_qps	The queries per second (QPS).
mysql_perf_redolog_size	The amount of redo logs that are generated on your computer. Unit: MB.
mysql_perf_slow_queries	The number of slow queries per second.
mysql_perf_sys_dir_size	The amount of storage space that is occupied by system files. Unit: MB.
mysql_perf_tmp_dir_size	The amount of storage space that is occupied by temporary files. Unit: MB.
mysql_perf_total_session	The average number of total connections.
mysql_perf_tps	The average number of transactions per second.

1.11.5. PolarDB-X 1.0

This topic describes the fields in the SQL audit log entries that are generated by PolarDB-X 1.0.

Field	Description
__topic__	The topic of the log entry. Valid values: drds_audit_log.
instance_id	The ID of the PolarDB-X 1.0 instance.
instance_name	The name of the PolarDB-X 1.0 instance.
owner_id	The ID of the Alibaba Cloud account that owns the PolarDB-X 1.0 instance.
region	The region where the PolarDB-X 1.0 instance resides.
db_name	The name of the PolarDB-X 1.0 database.
user	The username of the account that executes the SQL statement.
client_ip	The IP address of the database client that is used to access the PolarDB-X 1.0 instance.
client_port	The port number of the database client that is used to access the PolarDB-X 1.0 instance.
sql	The SQL statement that is executed.

Field	Description
trace_id	The trace ID of the SQL statement. If the SQL statement is executed in a transaction, the value of this field consists of a trace ID, a hyphen (-), and a number. For example, the value can be drdsabcdxyz-1.
sql_code	The hash value of the template SQL statement.
hint	The hint that specifies how to execute the SQL statement.
table_name	The name of the table that is specified in the SQL statement. If multiple tables are specified in the SQL statement, the names of the tables are separated by commas (,).
sql_type	The type of the SQL statement. Valid values: Select, Insert, Update, Delete, Set, Alter, Create, Drop, Truncate, Replace, and Other.
sql_type_detail	The name of the SQL parser.
response_time	The amount of time that is taken by the PolarDB-X 1.0 instance to respond. Unit: microseconds.
affect_rows	The number of rows that are returned for the SQL statement. If the SQL statement specifies an add, delete, or modify operation, the value of this field is the number of rows that are affected. If the SQL statement specifies a query operation, the value of this field is the number of rows that are returned.
fail	Indicates whether the SQL statement is successfully executed. Valid values: • 0: successful • 1: failed
sql_time	The time at which the SQL statement is executed.

1.11.6. SLB

This topic describes the fields of Layer 7 access logs in Server Load Balancer (SLB).

Log field	Description
owner_id	The ID of an Alibaba Cloud account.
region	The region where an instance resides.
instance_id	The ID of an instance.
instance_name	The name of an instance.

Log field	Description
network_type	The network type. Valid values: • VPC: a virtual private cloud (VPC) • Classic: the classic network
vpc_id	VPC ID
body_bytes_sent	The size of an HTTP message body that is sent to a client. Unit: bytes.
client_ip	The IP address of a client.
client_port	The client port.
host	The IP address of a server. The value is first obtained from the request parameters. If no value is obtained, the value is obtained from the host header field. If the value still cannot be obtained, the IP address of the backend server that processes the request is obtained as the field value.
http_host	The Host HTTP header in a request message.
http_referer	The Referer HTTP header in a request message that is received by the proxy.
http_user_agent	The User-Agent HTTP header in a request message that is received by the proxy.
http_x_forwarded_for	The X-Forwarded-For (XFF) HTTP header in a request message that is received by the proxy.
http_x_real_ip	The real IP address of a client.
read_request_time	The duration in which the proxy reads a request message. Unit: milliseconds.
request_length	The length of a request message. This field includes the start-line, HTTP headers, and HTTP body.
request_method	The request method.
request_time	The duration between the time when the proxy receives the first request message and the time when the proxy returns a response message. Unit: seconds.
request_uri	The URI of a request that is received by the proxy.
scheme	The protocol of an HTTP request. Valid values: HTTP and HTTPS.
server_protocol	The protocol of a request.
slb_vport	The listening port of an SLB instance.

Log field	Description
slbid	The ID of an SLB instance.
ssl_cipher	The used cipher suite.
ssl_protocol	The protocol that is used to establish an SSL connection.
status	The HTTP status code that is sent from the proxy.
tcpinfo_rtt	The RTT of TCP packets. Unit: microseconds.
time	The time when a log entry is recorded.
upstream_addr	The IP address and port number of the backend server.
upstream_response_time	The duration of the connection between the proxy and backend server. Unit: seconds.
upstream_status	The HTTP status code that is received by the proxy from the backend server.
vip_addr	The virtual IP address.
write_response_time	The period of time that is required by the proxy to respond to a write request. Unit: milliseconds.

1.11.7. ALB

This topic describes the fields in Layer 7 access logs of Application Load Balancer (ALB).

Field	Description
__topic__	The topic of the log. The value is fixed as alb_layer7_access_log.
body_bytes_sent	The number of bytes in the body of the HTTP response that is sent to the client.
client_ip	The IP address of the client.
host	The domain name or IP address of the server. By default, the value is obtained from request parameters. If no value is obtained from request parameters, the value is obtained from the host header field. If no value is obtained from request parameters or the host header field, the IP address of the backend server that processes the request is used as the value.
http_host	The host header field of the request.
http_referer	The URL of the source web page.
http_user_agent	The browser information of the client.

Field	Description
http_x_forwarded_for	The client IP address that is recorded after the request of the client is forwarded by the proxy.
http_x_real_ip	The real IP address of the client.
read_request_time	The time that is taken by the proxy to read the request. Unit: milliseconds.
request_length	The length of the request. The request line, request headers, and request body are all counted.
request_method	The request method.
request_time	The interval between the time when the proxy receives the first request and the time when the proxy returns a response. Unit: seconds.
request_uri	The URI of the request that is received by the proxy.
scheme	The schema of the request. Valid values: HTTP and HTTPS.
server_protocol	The HTTP version of the request that is received by the proxy. Example: HTTP/1.0 or HTTP/1.1.
slb_vport	The listening port of the SLB instance.
app_lb_id	The ID of the ALB instance.
ssl_cipher	The cipher suite that is used to establish an SSL connection. Example: ECDHE-RSA-AES128-GCM-SHA256.
ssl_protocol	The protocol that is used to establish an SSL connection. Example: TLSv1.2.
status	The HTTP status code that is sent by the proxy.
tcpinfo_rtt	The round-trip time (RTT) of the client TCP connection. Unit: microseconds.
time	The time when the log is generated.
upstream_addr	The IP address and port number of the backend server.
upstream_response_time	The interval between the time when the SLB instance connects to the backend server and the time when the SLB instance disconnects from the backend server after the required data is received.
upstream_status	The HTTP status code that is received by the proxy from the backend server.
vip_addr	The virtual IP address.

Field	Description
write_response_time	The time that is taken by the SLB proxy to send the request to the client after the SLB proxy receives the request from the backend server.
owner_id	The owner ID of the ALB instance.
region	The region where the ALB instance resides.
instance_name	The name of the ALB instance.
address_type	The address type of the ALB instance. Valid values: Intranet and Internet.
vpc_id	The ID of the Virtual Private Cloud (VPC) in which the ALB instance resides.

1.11.8. VPC

This topic describes the fields of Virtual Private Cloud (VPC) flow logs.

Field	Description
__topic__	The topic of the log. The value is fixed as flow_log.
version	The version of the flow log.
vswitch-id	The ID of the vSwitch to which the Elastic Network Interface (ENI) is bound.
vm-id	The ID of the Elastic Compute Service (ECS) instance to which the ENI is bound.
vpc-id	The ID of the VPC to which the ENI belongs.
account-id	The ID of the account.
eni-id	The ID of the ENI.
region	The region where the VPC resides.
srcaddr	The source address.
srcport	The source port.
dstaddr	The destination address.
dstport	The destination port.
protocol	The Internet Assigned Numbers Authority (IANA) protocol number of traffic. For more information, see Protocol Numbers.

Field	Description
direction	The direction of traffic. Valid values: • in: inbound traffic • out: outbound traffic
packets	The number of data packets.
bytes	The size of data packets.
start	The start time of the capture window.
end	The end time of the capture window.
log-status	The record status of the flow log. Valid values: • OK: Data is recorded. • NODATA: No inbound or outbound traffic is transmitted through the ENI during the capture window. • SKIPDATA: The status of some logs is not recorded during the capture window.
action	The action that is related to traffic. Valid values: • ACCEPT: Security groups allow the traffic to be recorded. • REJECT: Security groups do not allow the traffic to be recorded.

1.11.9. Bastionhost

This topic describes the fields in the operation log entries that are generated by Bastionhost.

Field	Description
__topic__	The topic of the log entry.
owner_id	The ID of the Alibaba Cloud account that owns the bastion host.
content	The content of the log entry.
event_type	The type of the recorded event. For more information, see Event types.
instance_id	The ID of the bastion host.
log_level	The severity level of the log entry.
resource_address	The address of the resource on which the recorded operation is performed.
resource_name	The name of the resource on which the recorded operation is performed.
result	The result of the recorded operation.

Field	Description
session_id	The ID of the session in which the recorded operation is performed.
user_client_ip	The IP address of the user who performs the recorded operation.
user_id	The ID of the user who performs the recorded operation.
user_name	The name of the user who performs the recorded operation.

Event types

Event type	Description
cmd.Command	Start Command Prompt.
cmd.Command.policy	Command processed based on control policies.
graph.Text	Text graph.
graph.Keyboard	Graphical keyboard event.
file.Upload	File upload.
file.Download	File download.
file.Rename	File renaming.
file.Delete	Object deletion.
file.DeleteDir	Directory deletion.
file.CreateDir	Directory creation.
login.CSLogin	User Client/Server (CS) logon.
Session.session	Session.

1.11.10. WAF

This topic describes the fields of access logs in Web Application Firewall (WAF).

Log field	Description
__topic__	The topic of the log. The value is fixed as waf_access_log.
owner_id	The ID of the Alibaba Cloud account.
account_action	The action that is performed on the client request after an account security rule is triggered. The value is fixed as *block*, which indicates that the request is blocked. For more information, see Description of the action field.

Log field	Description
account_rule_id	The ID of the account security rule that is triggered.
account_test	The protection mode that is used for the client request after an account security rule is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
acl_action	The action that is performed on the client request after a rule created for the blacklist or custom protection policy (ACL) feature is triggered. Valid values: block, captcha_strict, captcha, js, captcha_strict_pass, captcha_pass, and js_pass. For more information, see Description of the action field.
acl_rule_id	The ID of the rule that is triggered. The rule is created for the blacklist or custom protection policy (ACL) feature.
acl_rule_type	The type of the rule that is triggered. The rule is created for the blacklist or custom protection policy (ACL) feature. Valid values: • custom: indicates a rule that is created for the custom protection policy (ACL) feature. • blacklist: indicates a rule that is created for the blacklist feature.
acl_test	The protection mode that is used for the client request after a rule created for the blacklist or custom protection policy (ACL) feature is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
algorithm_rule_id	The ID of the rule that is triggered. The rule is created for the typical bot behavior identification feature.
antiscan_action	The action that is performed on the client request after a rule created for the scan protection feature is triggered. The value is fixed as *block*, which indicates that the request is blocked. For more information, see Description of the action field.
antiscan_rule_id	The ID of the rule that is triggered. The rule is created for the scan protection feature.

Log field	Description
antiscan_rule_type	The type of the rule that is triggered. The rule is created for the scan protection feature. Valid values: • highfreq: indicates a rule that blocks IP addresses from which web attacks are frequently initiated. • dirscan: indicates a rule that defends against path traversals. • scantools: indicates a rule that blocks the IP addresses of scanning tools. • collaborative: indicates a collaborative defense rule.
antiscan_test	The protection mode that is used for the client request after a rule created for the scan protection feature is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
block_action	The WAF protection feature that is triggered to block the request. **Notice** This field is no longer valid due to WAF upgrades. The final_plugin field replaces this field. If the block_action field is used in your services, replace the field with final_plugin at the earliest opportunity. • tmd: indicates the HTTP flood protection feature. • waf: indicates the web attack protection feature. • acl: indicates the custom protection policy feature. • deeplearning: indicates the Deep Learning Engine. • antiscan: indicates the scan protection feature. • antifraud: indicates the data risk control feature. • antibot: indicates the bot management feature.
body_bytes_sent	The number of bytes in the body of the client request.
bypass_matched_ids	The ID of the rule that is triggered to allow the client request. The rule can be a whitelist rule or a custom protection rule that allows the request. If multiple rules are triggered at the same time to allow the request, this field records the IDs of all the rules. Multiple IDs are separated by commas (,).
cc_action	The action that is performed on the client request after a rule created for the HTTP flood protection or custom protection policy (HTTP Flood Protection) feature is triggered. Valid values: block, captcha, js, captcha_pass, and js_pass. For more information, see Description of the action field.

Log field	Description
cc_blocks	Indicates whether the client request is blocked by the HTTP flood protection feature. Valid values: • 1: The request is blocked. • A different value: The request is allowed.
cc_rule_id	The ID of the rule that is triggered. The rule is created for the HTTP flood protection or custom protection policy (HTTP Flood Protection) feature.
cc_rule_type	The type of the rule that is triggered. The rule is created for the HTTP flood protection or custom protection policy (HTTP Flood Protection) feature. Valid values: • custom: indicates a custom protection rule (HTTP Flood Protection). • system: indicates an HTTP flood protection rule.
cc_test	The protection mode that is used for the client request after a rule created for the HTTP flood protection or custom protection policy (HTTP Flood Protection) feature is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
content_type	The type of the requested content.
deeplearning_action	The action that is performed on the client request after a rule created for the Deep Learning Engine is triggered. The value is fixed as *block*, which indicates that the request is blocked. For more information, see Description of the action field.
deeplearning_rule_id	The ID of the rule that is triggered. The rule is created for the Deep Learning Engine.
deeplearning_rule_type	The type of the rule that is triggered. The rule is created for the Deep Learning Engine. Valid values: • xss: indicates a rule that defends against cross-site scripting (XSS) attacks. • code_exec: indicates a rule that defends against specific attacks. The attacks exploit code execution vulnerabilities. • webshell: indicates a rule that defends against webshell uploads. • sqli: indicates a rule that defends against SQL injection. • lfilei: indicates a rule that defends against local file inclusion. • rfilei: indicates a rule that defends against remote file inclusion. • crlf: indicates a rule that defends against carriage return line feed (CRLF) injection. • other: indicates other protection rules.

Log field	Description
deeplearning_test	The protection mode that is used for the client request after a rule created for the Deep Learning Engine is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
dlp_rule_id	The ID of the rule that is triggered. The rule is created for the data leakage prevention feature.
dlp_test	The protection mode that is used for the client request after a rule created for the data leakage prevention feature is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
final_rule_type	The subtype of the rule that is applied to the client request. The rule is indicated by final_rule_id. For example, `final_plugin:waf` supports `final_rule_type:sqli` and `final_rule_type:xss`.
final_rule_id	The ID of the rule that is applied to the client request. The rule defines the action recorded in the final_action field.
final_action	The action that WAF performs on the client request. Valid values: block, captcha_strict, captcha, and js. For more information, see Description of the action field. If a request does not trigger a protection feature, the field is not recorded. For example, if a request matches a rule that allows the request or a client passes slider CAPTCHA verification or JavaScript verification, the field is not recorded. If a request triggers multiple protection features at the same time, the field is recorded, and the field includes only the action that is performed. The following actions are listed in descending order of priority: *block* (block), *captcha_strict* (strict slider CAPTCHA verification), *captcha* (common slider CAPTCHA verification), and *js* (JavaScript verification).

Log field	Description
final_plugin	The protection feature that performs the action specified by final_action on the client request. Valid values: • waf: indicates the Protection Rules Engine. • deeplearning: indicates the Deep Learning Engine. • dlp: indicates the data leakage prevention feature. • account: indicates the account security feature. • normalized: indicates the positive security model feature. • acl: indicates the blacklist or custom protection policy (ACL) feature. • cc: indicates the HTTP flood protection or custom protection policy (HTTP Flood Protection) feature. • antiscan: indicates the scan protection feature. • scene: indicates the scenario-specific configuration feature. • antifraud: indicates the data risk control feature. • intelligence: indicates the bot threat intelligence feature. • algorithm: indicates the typical bot behavior identification feature. • wxbb: indicates the app protection feature. To configure the preceding protection features, log on to the and choose **Protection Settings > Website Protection** in the left-side navigation pane. For more information about WAF protection features, see Overview of website protection. If a request does not trigger a protection feature, the field is not recorded. For example, if a request matches a rule that allows the request or a client passes slider CAPTCHA verification or JavaScript verification, the field is not recorded. If a request triggers multiple protection features at the same time, the field is recorded, and the field includes only the protection feature that performs the action specified by final_action.
host	The Host field of the request header. This field contains the domain name or IP address to access. The value of this field varies based on your service settings.
http_cookie	The Cookie field of the request header. This field contains the cookie information about the client.
http_referer	The Referer field of the request header. This field contains the source URL information about the request. If the request does not contain source URL information, the value of this field is a hyphen (-).
http_user_agent	The User-Agent field of the request header. This field contains information such as the identifier of the client browser or operating system.

Log field	Description
http_x_forwarded_for	The X-Forwarded-For (XFF) field of the request header. This field is used to identify the actual IP address of the client that is connected to the web server by using an HTTP proxy or a load balancing device.
https	Indicates whether the request is an HTTPS request. Valid values: • true: The request is an HTTPS request. • false: The request is an HTTP request.
matched_host	The domain name of the origin server that is matched by WAF for the request. A wildcard domain name may be matched. If no domain names are matched, the value of this field is a hyphen (-).
normalized_action	The action that is performed on the client request after a rule created for the positive security model feature is triggered. Valid values: block and continue. For more information, see Description of the action field.
normalized_rule_id	The ID of the rule that is triggered. The rule is created for the positive security model feature.
normalized_rule_type	The type of the rule that is triggered. The rule is created for the positive security model feature. Valid values: • User-Agent: indicates a User-Agent-based baseline rule. If the User-Agent field of a request header does not conform to the baseline, an attack may occur. This description applies to other rule types. • Referer: indicates a Referer-based baseline rule. • URL: indicates a URL-based baseline rule. • Cookie: indicates a cookie-based baseline rule. • Bod: indicates a request body-based baseline rule.
normalized_test	The protection mode that is used for the client request after a rule created for the positive security model feature is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
querystring	The query string in the client request. The query string refers to the part that follows the question mark (?) in the requested URL.
real_client_ip	The actual IP address of the client that initiates the request. WAF identifies the actual IP address based on the analysis of the request. If WAF cannot identify the actual IP address of the client, the value of this field is a hyphen (-). For example, if a proxy server is used or the IP field in the request header is invalid, WAF cannot identify the actual IP address of the client.

Log field	Description
region	The ID of the region where the WAF instance resides. Valid values: • cn: Chinese mainland • int: outside the Chinese mainland
remote_addr	The IP address that is used to connect to WAF. If WAF is directly connected to a client, this field records the actual IP address of the client. If a Layer 7 proxy, such as Content Delivery Network (CDN), is deployed in front of WAF, this field records the IP address of the proxy.
remote_port	The port that is used to connect to WAF. If WAF is directly connected to a client, this field records the port of the client. If a Layer 7 proxy, such as CDN, is deployed in front of WAF, this field records the port of the proxy.
request_length	The number of bytes in the client request. The request includes the request line, request headers, and request body. Unit: bytes.
request_method	The request method.
request_path	The requested relative path. The relative path refers to the part between the domain name and the question mark (?) in the requested URL. The relative path does not include the query string.
request_time_msec	The time that is taken by WAF to process the client request. Unit: milliseconds.
request_traceid	The unique identifier that is generated by WAF for the client request.
scene_action	The action that is performed on the client request after a rule created for scenario-specific configuration is triggered. Valid values: block, captcha, js, captcha_pass, and js_pass. For more information, see Description of the action field.
scene_id	The scenario ID of the rule that is triggered. The rule is created for scenario-specific configuration.
scene_rule_id	The ID of the rule that is triggered. The rule is created for scenario-specific configuration.

Log field	Description
scene_rule_type	The type of the rule that is triggered. The rule is created for scenario-specific configuration. Valid values: • bot_aialgo: indicates an intelligent protection rule. • js: indicates a rule that blocks script-based bots. • intelligence: indicates a rule that blocks attacks based on bot threat intelligence or data center blacklists. • sdk: indicates a rule that checks for abnormal signatures of SDK-integrated apps and abnormal device behaviors. • cc: indicates an IP address-based throttling rule or a custom session-based throttling rule.
scene_test	The protection mode that is used for the client request after a rule created for scenario-specific configuration is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
server_port	The requested destination port.
server_protocol	The protocol and version that is used by the origin server to respond to the request forwarded by WAF.
status	The HTTP status code that is returned by WAF to the client.
ssl_cipher	The cipher suite that is used in the client request.
ssl_protocol	The SSL or TLS protocol and version that are used in the client request.
time	The point in time at which the client request is initiated.
ua_browser	The name of the browser that initiates the request.
ua_browser_family	The family to which the browser belongs.
ua_browser_type	The type of the browser that initiates the request.
ua_browser_version	The version of the browser that initiates the request.
ua_device_type	The device type of the client that initiates the request.
ua_os	The operating system of the client that initiates the request.
ua_os_family	The family to which the operating system of the client belongs.

Log field	Description
upstream_addr	The back-to-origin addresses used by WAF. Each address is in the IP:Port format. Multiple addresses are separated by commas (,).
upstream_response_time	The time that is taken by the origin server to respond to the request. The request is forwarded by WAF. Unit: seconds. If a hyphen (-) is returned, the response timed out.
upstream_status	The status code that is returned by the origin server to WAF. If a hyphen (-) is returned, the request is not responded. For example, the request is blocked by WAF.
user_id	The ID of the Alibaba Cloud account to which the WAF instance belongs.
waf_action	The action that is performed on the client request after a rule created for the Protection Rules Engine is triggered. The value is fixed as *block*, which indicates that the request is blocked. For more information, see Description of the action field.
waf_test	The protection mode that is used for the client request after a rule created for the Protection Rules Engine is triggered. Valid values: • true: indicates the observation mode. In this mode, logs are recorded. However, protection actions, such as block, are not performed. • false: indicates the prevention mode. In this mode, WAF performs protection actions, such as block, on the request that matches the protection rule.
waf_rule_id	The ID of the rule that is triggered. The rule is created for the Protection Rules Engine.
waf_rule_type	The type of the rule that is triggered. The rule is created for the Protection Rules Engine. Valid values: • xss: indicates a rule that defends against XSS attacks. • code_exec: indicates a rule that defends against specific attacks. The attacks exploit code execution vulnerabilities. • webshell: indicates a rule that defends against webshell uploads. • sqli: indicates a rule that defends against SQL injection. • lfilei: indicates a rule that defends against local file inclusion. • rfilei: indicates a rule that defends against remote file inclusion. • crlf: indicates a rule that defends against CRLF injection. • other: indicates other protection rules.

1.11.11. Cloud Firewall

This topic describes the fields of traffic logs that are recorded for the Internet and virtual private cloud (VPC) firewalls in Cloud Firewall.

Fields of Internet firewall logs

Log field	Description
__topic__	The topic of a log. The value is fixed as cloudfirewall_access_log.
owner_id	The ID of an Alibaba Cloud account.
log_type	The type of log. The value is fixed as internet_log.
app_name	The type of application. Valid values include HTTPS, NTP, SIP, SMB, NFS, and DNS. If the type is unknown, the value Unknown is displayed.
direction	The direction of traffic. Valid values: • in: inbound traffic • out: outbound traffic
domain	The domain name of a destination server.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
end_time	The time at which a session ends. The value is a UNIX timestamp. Unit: seconds.
in_bps	The rate of inbound traffic. Unit: bit/s.
in_packet_bytes	The total size of inbound packets. Unit: bytes.
in_packet_count	The total number of inbound packets.
in_pps	The rate of inbound packets. Unit: packets per second (pps).
ip_protocol	The type of IP protocol. TCP and UDP are supported.
out_bps	The rate of outbound traffic. Unit: bit/s.
out_packet_bytes	The total size of outbound packets. Unit: bytes.
out_packet_count	The total number of outbound packets.
out_pps	The rate of outbound packets. Unit: pps.
region_id	The ID of the region from which access traffic originates.

Log field	Description
rule_result	The result of how an access control policy processes access traffic. Valid values: • pass: Access traffic is allowed to pass Cloud Firewall. • alert: An alert is triggered when access traffic passes Cloud Firewall. • drop: Access traffic is blocked.
src_ip	The IP address of a source server.
src_port	The source port. A host sends data from this port.
start_time	The time at which a session starts. The value is a UNIX timestamp. Unit: seconds.
start_time_min	The time at which a session starts. The value is a UNIX timestamp. The value is rounded up to the next minute. Unit: seconds.
tcp_seq	The sequence number of a TCP segment.
total_bps	The total rate of inbound and outbound traffic. Unit: bit/s.
total_packet_bytes	The total size of inbound and outbound packets. Unit: bytes.
total_packet_count	The total number of inbound and outbound packets.
total_pps	The total rate of inbound and outbound packets. Unit: pps.
src_private_ip	The private IP address of a source server.
vul_level	The risk level of a vulnerability. Valid values: • 1: low • 2: medium • 3: high
url	The URL that is accessed.
acl_rule_id	The ID of an access control list (ACL) policy that is matched.
ips_rule_id	The ID of an intrusion prevention system (IPS) policy that is matched.
ips_ai_rule_id	The ID of an intelligent policy that is matched.
ips_rule_name	The Chinese name of an IPS policy that is matched.
ips_rule_name_en	The English name of an IPS policy that is matched.
attack_type_name	The Chinese name of an attack type.
attack_type_name_en	The English name of an attack type.
proxy_acl_rule_id	The ID of an ACL policy that is matched by forward proxies.

Fields of VPC firewall logs

Log field	Description
__topic__	The topic of a log. The value is fixed as cloudfirewall_vpc_log.
log_type	The type of log. The value is fixed as vpc_firewall_log.
aliuid	The ID of an Alibaba Cloud account.
app_name	The type of application. Valid values include HTTPS, NTP, SIP, SMB, NFS, and DNS. If the type is unknown, the value Unknown is displayed.
domain	The domain name of a destination server.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
dst_region	The ID of the region for which access traffic is destined.
dst_network_instance_id	The ID of the instance for which access traffic is destined. The instance may be a VPC, virtual border router (VBR), or Cloud Connect Network (CCN).
end_time	The time at which a session ends. The value is a UNIX timestamp. Unit: seconds.
firewall_id	The ID of a VPC firewall. • If Cloud Enterprise Network (CEN) is used, the ID of the CEN instance is displayed. Example: cen-6srj4tvjjovhbc. • If Express Connect is used, the ID of the firewall instance is displayed. Example: vfw-123.
in_bps	The rate of inbound traffic. Unit: bit/s.
in_packet_bytes	The total size of inbound packets. Unit: bytes.
in_packet_count	The total number of inbound packets.
in_pps	The rate of inbound packets. Unit: pps.
ip_protocol	The type of IP protocol. TCP and UDP are supported.
out_bps	The rate of outbound traffic. Unit: bit/s.
out_packet_bytes	The total size of outbound packets. Unit: bytes.
out_packet_count	The total number of outbound packets.
out_pps	The rate of outbound packets. Unit: pps.

Log field	Description
rule_result	The result of how an access control policy processes access traffic. Valid values: • pass: Access traffic is allowed to pass Cloud Firewall. • alert: An alert is triggered when access traffic passes Cloud Firewall. • drop: Access traffic is blocked.
src_ip	The IP address of a source server.
src_port	The source port.
src_region	The ID of the region from which access traffic originates.
src_network_instance_id	The ID of the instance from which access traffic originates. The instance may be a VPC, VBR, or CCN.
start_time	The time at which a session starts. The value is a UNIX timestamp. Unit: seconds.
start_time_min	The time at which a session starts. The value is a UNIX timestamp. The value is rounded up to the next minute. Unit: seconds.
tcp_seq	The sequence number of a TCP segment.
total_bps	The total rate of inbound and outbound traffic. Unit: bit/s.
total_packet_bytes	The total size of inbound and outbound packets. Unit: bytes.
total_packet_count	The total number of inbound and outbound packets.
total_pps	The total rate of inbound and outbound packets. Unit: pps.
vul_level	The risk level of a vulnerability. Valid values: • 1: low • 2: medium • 3: high
ips_rule_name	The Chinese name of an IPS policy that is matched.
ips_rule_name_en	The English name of an IPS policy that is matched.
attack_type_name	The Chinese name of an attack type.
attack_type_name_en	The English name of an attack type.

1.11.12. Anti-DDoS

This topic describes the fields of access logs in Anti-DDoS Pro, Anti-DDoS Premium, and Anti-DDoS Origin.

Anti-DDoS Pro

Log field	Description
__topic__	The topic of a log entry. Valid value: ddoscoo_access_log.
owner_id	The ID of an Alibaba Cloud account.
body_bytes_sent	The size of a request body. Unit: bytes.
cc_action	The action that is performed based on an HTTP flood protection policy. The action can be none, challenge, pass, close, captcha, wait, or login.
cc_phase	The HTTP flood protection policy that is matched. The policy can be seccookie, server_ip_blacklist, static_whitelist, server_header_blacklist, server_cookie_blacklist, server_args_blacklist, or qps_overmax.
cc_blocks	Indicates whether a request is blocked by an HTTP flood protection policy. Valid values: • If the value is 1, the request is blocked. • If the value is not 1, the request is passed.
content_type	The content type of a request.
host	The origin server.
http_cookie	The Cookie HTTP header.
http_referer	The Referer HTTP header. If an HTTP header does not contain a referer, a hyphen (-) is displayed.
http_user_agent	The User-Agent HTTP header.
http_x_forwarded_for	The IP address of an upstream user. The IP address is forwarded by a proxy server.
https	Indicates whether a request is an HTTPS request. Valid values: • true: The request is an HTTPS request. • false: The request is an HTTP request.
isp_line	The information of an Internet service provider (ISP) line, for example, BGP, China Telecom, or China Unicom.
matched_host	The matched origin server, which can be a wildcard domain name. If no origin server is matched, a hyphen (-) is displayed.
real_client_ip	The real IP address of a client. If no real IP address can be obtained, a hyphen (-) is displayed.
remote_addr	The IP address of a client that sends an access request.

Log field	Description
remote_port	The port number of a client that sends an access request.
request_length	The size of a request. Unit: bytes.
request_method	The HTTP method of a request.
request_time_msec	The duration in which a request is processed. Unit: microseconds.
request_uri	The uniform resource identifier (URI) of a request.
server_name	The name of a matched server. If no server name is matched, default is displayed.
status	The HTTP status code.
time	The time when a request is sent.
ua_browser	The browser.
ua_browser_family	The family to which a browser belongs.
ua_browser_type	The type of a browser.
ua_device_type	The type of a client.
ua_os	The operating system of a client.
ua_os_family	The family of the operating system that runs on a client.
upstream_addr	The list of back-to-origin IP addresses. Each IP address is in the IP:Port format. Multiple IP addresses are separated by commas (,).
upstream_ip	The real IP address of an origin server.
upstream_response_time	The response time of a back-to-origin process. Unit: seconds.
upstream_status	The HTTP status code of a back-to-origin request.

Anti-DDoS Premium

Log field	Description
__topic__	The topic of a log entry. Valid value: ddosdip_access_log.
owner_id	The ID of an Alibaba Cloud account.
body_bytes_sent	The size of a request body. Unit: bytes.

Log field	Description
cc_action	The action that is performed based on an HTTP flood protection policy. The action can be none, challenge, pass, close, captcha, wait, or login.
cc_phase	The HTTP flood protection policy that is matched. The policy can be seccookie, server_ip_blacklist, static_whitelist, server_header_blacklist, server_cookie_blacklist, server_args_blacklist, or qps_overmax.
cc_blocks	Indicates whether a request is blocked by an HTTP flood protection policy. Valid values: • If the value is 1, the request is blocked. • If the value is not 1, the request is passed.
content_type	The content type of a request.
host	The origin server.
http_cookie	The Cookie HTTP header.
http_referer	The Referer HTTP header. If an HTTP header does not contain a referer, a hyphen (-) is displayed.
http_user_agent	The User-Agent HTTP header.
http_x_forwarded_for	The IP address of an upstream user. The IP address is forwarded by a proxy server.
https	Indicates whether a request is an HTTPS request. Valid values: • true: The request is an HTTPS request. • false: The request is an HTTP request.
isp_line	The information of an ISP line, for example, BGP, China Telecom, or China Unicom.
matched_host	The matched origin server, which can be a wildcard domain name. If no origin server is matched, a hyphen (-) is displayed.
real_client_ip	The real IP address of a client. If no real IP address can be obtained, a hyphen (-) is displayed.
remote_addr	The IP address of a client that sends an access request.
remote_port	The port number of a client that sends an access request.
request_length	The size of a request. Unit: bytes.
request_method	The HTTP method of a request.
request_time_msec	The duration in which a request is processed. Unit: microseconds.
request_uri	The URI of a request.

Log field	Description
server_name	The name of a matched server. If no server name is matched, default is displayed.
status	The HTTP status code.
time	The time when a request is sent.
ua_browser	The browser.
ua_browser_family	The family to which a browser belongs.
ua_browser_type	The type of a browser.
ua_device_type	The type of a client.
ua_os	The operating system of a client.
ua_os_family	The family of the operating system that runs on a client.
upstream_addr	The list of back-to-origin IP addresses. Each IP address is in the IP:Port format. Multiple IP addresses are separated by commas (,).
upstream_ip	The real IP address of an origin server.
upstream_response_time	The response time of a back-to-origin process. Unit: seconds.
upstream_status	The HTTP status code of a back-to-origin request.

Anti-DDoS Origin

Log field	Description
__topic__	The topic of a log entry. Valid value: ddosbgp_access_log.
data_type	The type of a log entry.
event_type	The type of an event.
ip	The IP address from which the request is sent.
subnet	The CIDR block of the instance that is rerouted.
event_time	The date when an event occurs, for example, 2020-01-01.
qps	The number of queries per second when an event occurs.
pps_in	The rate of inbound traffic when an event occurs. Unit: packets per second (pps).

Log field	Description
new_con	The new connection that is established when an event occurs.
kbps_in	The rate of inbound traffic when an event occurs. Unit: bit/s.
instance_id	The ID of an instance.
time	The time when a log is generated, for example, 2020-07-17 10:00:30.
destination_ip	The IP address of a destination server.
port	The destination port.
total_traffic_in_bps	The rate of total inbound traffic. Unit: bit/s.
total_traffic_drop_bps	The rate of total inbound traffic that is dropped. Unit: bit/s.
total_traffic_in_pps	The rate of total inbound traffic. Unit: pps.
total_traffic_drop_pps	The rate of total inbound traffic that is dropped. Unit: pps.
pps_types_in_tcp_pps	The rate of inbound TCP traffic that is measured by protocol. Unit: pps.
pps_types_in_udp_pps	The rate of inbound UDP traffic that is measured by protocol. Unit: pps.
pps_types_in_icmp_pps	The rate of inbound ICMP traffic that is measured by protocol. Unit: pps.
pps_types_in_syn_pps	The rate of inbound SYN traffic that is measured by protocol. Unit: pps.
pps_types_in_ack_pps	The rate of inbound ACK traffic that is measured by protocol. Unit: pps
user_id	The ID of an Alibaba Cloud account.

1.11.13. Security Center

This topic describes the fields of Security Center logs. Security Center logs include network logs, security logs, and host logs.

Network logs

- DNS logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-log-dns.
owner_id	The ID of an Alibaba Cloud account.

Log field	Description
additional	The fields in the additional section. Multiple values are separated by vertical bars (|).
additional_num	The number of fields in the additional section.
answer	The DNS responses. Multiple values are separated by vertical bars (|).
answer_num	The number of DNS responses.
authority	The fields in the authority section.
authority_num	The number of fields in the authority section.
client_subnet	The subnet where a client resides.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
in_out	The direction of data flows. Valid values: ○ in: inbound ○ out: outbound
qid	The ID of a query.
qname	The domain name that is queried.
qtype	The type of a resource that is queried.
query_datetime	The timestamp of a query. Unit: milliseconds.
rcode	The code of a response.
region	The ID of a source region. Valid values: ○ 1: China (Beijing) ○ 2: China (Qingdao) ○ 3: China (Hangzhou) ○ 4: China (Shanghai) ○ 5: China (Shenzhen) ○ 6: Others
response_datetime	The time when a response is returned.
src_ip	The IP address of a source server.
src_port	The source port.

- Local DNS logs

Log field	Description
__topic__	The topic of a log entry. Valid value: local-dns.
owner_id	The ID of an Alibaba Cloud account.
answer_rdata	The DNS responses. Multiple values are separated by vertical bars (|).
answer_ttl	The time-to-live (TTL) of resource records in DNS responses. Multiple values are separated by vertical bars (|).
answer_type	The types of resource records in DNS responses. Multiple values are separated by vertical bars (|).
anwser_name	The domain names in DNS responses. Multiple values are separated by vertical bars (|).
dest_ip	The IP address of a destination server.
dest_port	The destination port.
group_id	The ID of the group to which a host belongs.
hostname	The hostname.
id	The ID of a query.
instance_id	The ID of an instance.
internet_ip	The public IP address of a host.
ip_ttl	The TTL of the data packets that are sent by a host.
query_name	The domain name that is queried.
query_type	The type of a resource that is queried.
src_ip	The IP address of a source server.
src_port	The source port.
time	The timestamp of a query. Unit: seconds.
time_usecond	The response time. Unit: microseconds.
tunnel_id	The ID of a DNS tunnel.

- Network session logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-log-session.
owner_id	The ID of an Alibaba Cloud account.

Log field	Description
asset_type	The type of an associated Alibaba Cloud service, for example, ECS, SLB, or ApsaraDB RDS.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
proto	The type of a transport layer protocol, for example, TCP or UDP.
session_time	The duration of a session.
src_ip	The IP address of a source server.
src_port	The source port.

- Web logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-log-http.
owner_id	The ID of an Alibaba Cloud account.
content_length	The content length of an HTTP request message.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
host	The hostname of a web server.
jump_location	The IP address of an HTTP redirect.
method	The HTTP request method.
referer	The Referer HTTP header. This field includes the address of the web page that sends a request.
request_datetime	The time when a request is sent.
ret_code	The HTTP status code.
rqs_content_type	The content type of an HTTP request message.
rsp_content_type	The content type of an HTTP response message.
src_ip	The IP address of a source server.
src_port	The source port.
uri	The URI of a request.

Log field	Description
user_agent	The user agent of a client that sends a request.
x_forward_for	The X-Forwarded-For (XFF) HTTP header.

Security logs

- Vulnerability logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-vul-log.
owner_id	The ID of an Alibaba Cloud account.
name	The name of a vulnerability.
alias_name	The alias of a vulnerability.
op	The action that is performed on a vulnerability. Valid values: ○ new: detects a new vulnerability. ○ verify: verifies a vulnerability. ○ fix: fixes a vulnerability.
status	The status of a vulnerability. For more information, see Status codes of security logs.
tag	The tag of a vulnerability, for example, oval, system, or cms. This field is used to distinguish different emergency (EMG) vulnerabilities.
type	The type of a vulnerability. Valid values: ○ sys: Windows vulnerability ○ cve: Linux vulnerability ○ cms: Web CMS vulnerability ○ EMG: emergency vulnerability
uuid	The universally unique identifier (UUID) of a client.

- Baseline logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-hc-log.
owner_id	The ID of an Alibaba Cloud account.
level	The level of a baseline.

Log field	Description
op	The action that is performed on a baseline. Valid values: ○ new: detects a new baseline. ○ verify: verifies a baseline.
risk_name	The name of a baseline risk.
status	The status of a baseline. For more information, see Status codes of security logs.
sub_type_alias	The subtype alias of a baseline.
sub_type_name	The subtype of a baseline.
type_name	The type of a baseline. For more information, see Types and subtypes of baselines.
type_alias	The type alias of a baseline.
uuid	The UUID of a client.
check_item	The name of a check item.
check_level	The level of a check item.
check_type	The type of a check item.

Types and subtypes of baselines

type_name	sub_type_name
system	baseline
weak_password	postsql_weak_password
database	redis_check
account	system_account_security
account	system_account_security
weak_password	mysq_weak_password
weak_password	ftp_anonymous
weak_password	rdp_weak_password
system	group_policy
system	register
account	system_account_security

type_name	sub_type_name
weak_password	sqlserver_weak_password
system	register
weak_password	ssh_weak_password
weak_password	ftp_weak_password
cis	centos7
cis	tomcat7
cis	memcached-check
cis	mongodb-check
cis	ubuntu14
cis	win2008_r2
system	file_integrity_mon
cis	linux-httpd-2.2-cis
cis	linux-docker-1.6-cis
cis	SUSE11
cis	redhat6
cis	bind9.9
cis	centos6
cis	debain8
cis	redhat7
cis	SUSE12
cis	ubuntu16

Status codes of security logs

Status code	Description
1	Unfixed.
2	Fix failed.
3	Rollback failed.

Status code	Description
4	Fixing.
5	Rolling back.
6	Verifying.
7	Fixed.
8	Fixed. Waiting for a restart.
9	Rollback succeeded.
10	Ignored.
11	Rollback succeeded. Waiting for a restart.
12	No longer exists.
20	Expired.

- Security alert logs

Log field	Description
__topic__	The topic of a log entry. Valid value: sas-security-log.
data_source	The data source. For more information, see Values of the data_source field in security alert logs.
level	The severity level of an alert.
name	The name of an alert.
op	The action that is performed on an alert. Valid values: ○ new: An alert is triggered. ○ dealing: An alert is being processed.
status	The status of an alert. For more information, see Status codes of security logs.
uuid	The UUID of a client.
detail	The details of an alert.
unique_info	The unique identifier of an alert for a single server.

Values of the data_source field in security alert logs

Value	Description

Value	Description
aegis_suspicious_event	Server exceptions
aegis_suspicious_file_v2	Webshell
aegis_login_log	Suspicious logons
security_event	Security Center exceptions

Host logs

- Process startup logs

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-log-process.
uuid	The UUID of a client.
ip	The IP address of a client.
cmdline	The full command line that starts a process.
username	The username.
uid	The ID of a user.
pid	The ID of a process.
filename	The name of a process file.
filepath	The full path of a process file.
groupname	The name of a user group.
ppid	The ID of a parent process.
pfilename	The name of a parent process file.
pfilepath	The full path of a parent process file.
cmd_chain	The process chain.
containerhostname	The hostname of a container.
containerpid	The process ID of a container.
containerimageid	The ID of an image.
containerimagename	The name of an image.
containername	The name of a container.

Log field	Description
containerid	The ID of a container.
cwd	The current working directory (CWD) of a running process.

- Process snapshot logs

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-snapshot-process.
owner_id	The ID of an Alibaba Cloud account.
uuid	The UUID of a client.
ip	The IP address of a client.
cmdline	The full command line that starts a process.
pid	The ID of a process.
name	The name of a process file.
path	The full path of a process file.
md5	The MD5 hash of a process file. If the process file exceeds 1 MB, the MD5 hash is not calculated.
pname	The name of a parent process file.
start_time	The time when a process starts. This field is a built-in field.
user	The username.
uid	The ID of a user.

- Logon logs

  The logon attempts within 1 minute are recorded in one log entry.

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-log-login.
owner_id	The ID of an Alibaba Cloud account.
uuid	The UUID of a client.
ip	The IP address of a client.
warn_ip	The IP address of a source server.
warn_port	The logon port.

Log field	Description
warn_type	The type of a logon. Valid values: SSHLOGIN, RDPLOGIN, and IPCLOGIN.
warn_user	The logon username.
warn_count	The number of logon attempts. In this example, the value 3 indicates that two logon requests are sent 1 minute before the current logon.

- Brute-force cracking logs

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-log-crack.
owner_id	The ID of an Alibaba Cloud account.
uuid	The UUID of a client.
ip	The IP address of a client.
warn_ip	The IP address of a source server.
warn_port	The logon port.
warn_type	The type of a logon. Valid values: SSHLOGIN, RDPLOGIN, and IPCLOGIN.
warn_user	The logon username.
warn_count	The number of failed logon attempts.

- Network connection logs

  The changes in network connections are collected on the host every 10 seconds to 1 minute.

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-log-network.
owner_id	The ID of an Alibaba Cloud account.
uuid	The UUID of a client.
ip	The IP address of a client.
src_ip	The IP address of a source server.
src_port	The source port.
dst_ip	The IP address of a destination server.
dst_port	The destination port.
proc_name	The name of a process.

Log field	Description
proc_path	The path of a process file.
proto	The protocol that is used to establish a network connection.
status	The connection status. For more information, see Status codes of network connections.

Status codes of network connections

Status code	Description
1	closed
2	listen
3	syn send
4	syn recv
5	establisted
6	close wait
7	closing
8	fin_wait1
9	fin_wait2
10	time_wait
11	delete_tcb

- Port listening snapshot logs

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-snapshot-port.
owner_id	The ID of an Alibaba Cloud account.
uuid	The UUID of a client.
ip	The IP address of a client.
proto	The protocol that is used by a listener.
src_ip	The IP address that is listened on.
src_port	The port that is listened on.

Log field	Description
pid	The ID of a process.
proc_name	The name of a process.

- Account snapshot logs

Log field	Description
__topic__	The topic of a log entry. Valid value: aegis-snapshot-host.
owner_id	The ID of an Alibaba Cloud account.
name	The name of a vulnerability.
alias_name	The alias of a vulnerability.
op	The action that is performed on a vulnerability. Valid values: ○ new: detects a new vulnerability. ○ verify: verifies a vulnerability. ○ fix: fixes a vulnerability.
status	The connection status. For more information, see Status codes of network connections.
tag	The tag of a vulnerability, for example, oval, system, or cms. This field is used to distinguish different emergency (EMG) vulnerabilities.
type	The type of a vulnerability. Valid values: ○ sys: Windows vulnerability ○ cve: Linux vulnerability ○ cms: Web CMS vulnerability ○ EMG: emergency vulnerability
uuid	The UUID of a client.

1.11.14. API Gateway

This topic describes the fields of access logs in API Gateway.

Log field	Description
owner_id	The ID of the Alibaba Cloud account to which an API belongs.
apiGroupUid	The ID of the group to which an API belongs.
apiGroupName	The name of the group to which an API belongs.
apiUid	API ID

Log field	Description
apiName	The name of an API.
apiStageUid	The stage ID of an API.
apiStageName	The stage name of an API.
httpMethod	The HTTP request method.
path	The uniform resource identifier (URI) of a request.
domain	The domain name of a resource for which an API request is sent.
statusCode	The HTTP status code.
errorMessage	The error message that is returned.
appId	The ID of the application from which an API request is sent.
appName	The name of the application from which an API request is sent.
clientIp	The IP address of a client that sends an API request.
exception	The specific error message that is returned by a backend server.
region	The ID of a region.
requestHandleTime	The time when an API request is sent. The time is in Greenwich Mean Time (GMT).
requestId	The ID of an API request. The ID is globally unique.
requestSize	The size of an API request. Unit: bytes.
responseSize	The size of a response message. Unit: bytes.
serviceLatency	The response latency of a backend server. Unit: milliseconds.

1.11.15. Apsara File Storage NAS

This topic describes the fields of access logs in Apsara File Storage NAS.

Log field	Description
owner_id	The ID of an Alibaba Cloud account.
ArgIno	The inode number of a file system.
AuthRc	The authorization code that is returned.
NFSProtocolRc	The return code of the Network File System (NFS) protocol.

Log field	Description
OpList	The procedure number of the NFSv4 protocol.
Proc	The procedure number of the NFSv3 protocol.
RWSize	The size of read and write data. Unit: bytes.
RequestId	The ID of a request.
ResIno	The inode number of a resource that is looked up.
SourceIp	The IP address of a client.
Vers	The version number of the NFS protocol.
Vip	The IP address of a server.
Volume	The ID of a file system.
microtime	The time when a request is sent. Unit: microseconds.

1.11.16. CSB App Connect

This topic describes the fields of operation logs in Cloud Service Bus (CSB) App Connect.

Log field	Description
__topic__	The topic of a log entry. Valid value: appconnect_oplog.
uid	The ID of an Alibaba Cloud account.
execution_id	The ID of a request or the ID of an execution.
status	The execution status of an integration flow. Valid values: begin and done.
flow_name	The name of an integration flow.
step	The name of a step in an integration flow. The name is the unique identifier of the step.
id	The ID of a step. The ID is the unique index that is used to implement each integration flow. The ID can be decoded by using the stepTime timestamp field. A step can be executed multiple times in scenarios where loops are included. The step has the same name but different IDs.
type	The type of a step.
duration	The duration of a step. Unit: nanoseconds.

Log field	Description
message	The output of a step. The output is in the string format.
step_time	The time when a step is executed by an integration flow.
container_ip	The IP address of a pod.
integration_name	The name of a pod.
failed	Indicates whether a step is implemented.

1.12. FAQ and troubleshooting

This topic describes some common errors of Log Audit Service and the troubleshooting methods for the errors. This topic also provides answers to some frequently asked questions about Log Audit Service.

Common errors and troubleshooting

Error type	Error message	Cause	Solution
	`LogException{httpCode=-1 errorCode='IllegalResourceDirectoryAccounts' errorMessage='IllegalResourceDirectoryAccounts: account not ResourceDirectory master or admin user' requestId=''`	The current account is a standard central account. You cannot use the current account to configure multi-account collection in resource directory mode. You can configure multi-account collection in resource directory mode only when the central account is the management account or a delegated administrator account of your resource directory. For more information, see Configure multi-account collection.	Configure multi-account collection in custom authentication mode. For more information, see Custom authentication mode.

Error type	Error message	Cause	Solution
Account configuration	`LogException{httpCode=-1 errorCode='IllegalAction.MultiAccountsIllegal' errorMessage='IllegalAction: the multi_account: 1234567*** may be already configured by other central account or contain central account' requestId=''}`	Conflicts occur in the multi-account collection configuration. • If you have used Account A as the central account to activate Log Audit Service, you cannot add Central Account A to the resource directory of Central Account C as a member. • If Account A is a member in the resource directory of Central Account B, you cannot add Account A to the resource directory of Central Account C as a member.	If you need to add `1234567***` to the resource directory of the current central account as a member, use one of the following methods: • If `1234567***` is a central account, delete all Log Audit Service resources from `1234567***` and add 1234567*** to the resource directory of the current central account. For more information, see Delete Log Audit Service resources. • If `1234567***` is a member in the resource directory of a central account, delete the multi-account collection configuration for `1234567***` from the central account and then add 1234567*** to the resource directory of the current central account.
	When an account is added to the resource directory of a central account as a member, the EtlMetaAlreadyExist error occurs.	The account has been added to the resource directory of the central account or a different central account as a member. In addition, the projects of Log Audit Service were deleted when log collection is enabled for cloud services. This does not comply with the rule. Before you can delete the projects of Log Audit Service, you must disable log collection for all cloud services. For more information, see Delete Log Audit Service resources. As a result, the system cannot apply a new collection configuration.	Submit a ticket to contact the Log Service team.

Error type	Error message	Cause	Solution
	`The role not exists: acs:ram::123456******:role/sls-audit-service-monitor.`	The permissions of the sls-audit-service-monitor role within the member `123456******` are deleted or tampered with.	Reconfigure the permissions for the sls-audit-service-monitor role. For more information, see Use a custom policy to authorize Log Service to collect and synchronize logs.
			Grant the required permissions to the RAM user. The following examples show the policies that you can use to grant the permissions. For more information, see Create a custom policy and Grant permissions to a RAM user. • Grant the RAM user the permissions to view and configure data in Log Audit Service.

Error type	Error message	Cause	Solution

Permission configuration	`Permission denied, action: log:CreateApp,resource: app/audit` or `Permission denied, action: log:GetApp,resource: app/audit`	The RAM user that is used does not have the required operation permissions on Log Audit Service.	<pre>{
			"Version": "1",
			"Statement": [
			{
			"Effect": "Allow",
			"Action": [
			"log:GetApp",
			"log:CreateApp"
			],
			"Resource": [
			"acs:log:*:*:app/audit"
			]
			},
			{
			"Effect": "Allow",
			"Action": [
			"log:Get*",
			"log:List*",
			"log:CreateJob",
			"log:UpdateJob",
			"log:CreateProject"
			],
			"Resource": [
			"acs:log:*:*:project/slsaudit-*"
			]
			}
			]
			}</pre>
			• Grant the RAM user only the permissions to view data in Log Audit Service.

Error type	Error message	Cause	Solution

			{
			"Version": "1",
			"Statement": [
			{
			"Effect": "Allow",
			"Action": [
			"log:GetApp"
			],
			"Resource": [
			"acs:log:*:*:app/audit"
			]
			},
			{
			"Effect": "Allow",
			"Action": [
			"log:Get*",
			"log:List*"
			],
			"Resource": [
			"acs:log:*:*:project/slsaudit-*"
			]
			}
			]
			}

Quota limit	init_sls_assets failed because of ServerException [HTTP Status: 400 Error:DashboardError LogException{httpCode=403 errorCode='ExceedQuota' requestId='622854********'}	A quota is exceeded. Each type of basic resource provided by Log Service has a quota. The basic resources include shards. For more information, see Basic resources.	If the ExceedQuota error occurs in a project of Log Audit Service, submit a ticket to scale out basic resources.

Error type	Error message	Cause	Solution
	`init_sls_assets failed because of ServerException [HTTP Status: 400 Error:CreateProjectFailed Account 123456***** most has 50 project`	The total number of projects within a single account exceeds the quota. Each type of basic resource provided by Log Service has a quota. The basic resources include projects and shards. For more information, see Basic resources.	If the total number of Log Audit Service projects within a single account exceeds the quota, submit a ticket to scale out basic resources.
	When you change the log retention period, the following message appears: `This Logstore is dedicated to the Log Audit Service application. To modify the Logstore settings such as the data retention period, go to the Global Configurations page of Log Audit Service.`	None	On the **Global Configurations** page of Log Audit Service, change the log retention period of a Logstore and the log retention period for the hot storage of a Logstore.
	When you change the indexes of a Logstore, the following message appears: `This Logstore is dedicated to the Log Audit Service application. You cannot modify the index attributes of the Logstore or disable indexing.`	Log Audit Service Logstores may be associated with built-in dashboards and alerts. If a Log Audit Service Logstore is associated with built-in dashboards and alerts, you cannot perform index management operations on the Logstore. For example, you cannot change indexes or disable indexing. **Note** You can change the indexes of Logstores for Container Service for Kubernetes (ACK). However, you cannot disable indexing for the Logstores.	None

Error type	Error message	Cause	Solution
Resource audit	When you delete a Logstore, the following message appears: `Operation failed: Insufficient permissions`.	Log Audit Service Logstores may be associated with built-in dashboards and alerts. Therefore, you cannot separately delete Log Audit Service Logstores.	• If you want to delete logs from a Logstore, you can change the log retention period for the Logstore to a minimum value on the **Global Configurations** page of Log Audit Service. Then, save the change and disable log collection. Log Service automatically deletes the logs when the log retention period ends. • If you need to delete a Logstore, you can delete the project to which the Logstore belongs. For more information, see Delete Log Audit Service resources.
	LogException{httpCode=-1 errorCode='DeleteFailed'	The central project of Log Audit Service is supported only in some regions. If you use other methods to specify an unsupported region such as the China (Chengdu) region for the central project of Log Audit Service, the system creates a project named `slsaudit-center-${uid}-${Region}` in this region. However, you cannot use Log Audit Service in the region or switch the region over to a supported region.	Delete the (`slsaudit-center-${uid}-${Region}`) project by using a CLI or API operation.

Error type	Error message	Cause	Solution
	errorCode= DeleteFailed message='delete auditJob error' requestId=''}	Note The following regions are supported: • Chinese mainland: China (Qingdao), China (Beijing), China (Hohhot), China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Hong Kong) • Outside the Chinese mainland: Singapore (Singapore), Japan (Tokyo), Germany (Frankfurt), and Indonesia (Jakarta)	Then, select a supported region. For more information about how to delete a project, see What to do next.

FAQ

- Before I disable log collection, I changed the log retention period. However, the change does not take effect. For example, before I disable log collection for the Layer 7 access logs of Server Load Balancer (SLB), I changed the log retention period to one day to delete existing logs. However, the change does not take effect, and the log retention period remains 180 days. Why?

  If you change the log retention period before you disable log collection, you must save the change before you disable log collection. Otherwise, the change does not take effect. To change the log retention period, perform the following operations:

  i. On the **Global Configurations** page of Log Audit Service, click **Modify**.
  ii. Change the log retention period for your log type and click **Save**.

  > Note Make sure that log collection is enabled when you save the change. After you save the change, wait 1 minute and click Modify again.

  iii. Click **Modify** again.
  iv. Disable log collection for your log type and click **Save**.
- How do I view the collection status of logs?

Choose **Access to Cloud Products > Status Dashboard** of Log Audit Service to view the collection status of logs.

- The system prompts that my account does not have the required permissions or the AccessKey pair of my account is invalid. What do I do?

  Check whether permissions are correctly configured for your account. If Log Service and the cloud service from which logs are collected belong to the same account, follow the instructions provided in Initially configure Log Audit Service. If Log Service and the cloud service belong to different accounts, follow the instructions provided in Use a custom policy to authorize Log Service to collect and synchronize logs. For example, if the **ReadOnlyAccess** policy under **System Policy** is not attached to the sls-audit-service-monitor role, this issue occurs.

- The system prompts that a required feature is not enabled for my account. What do I do?

  Enable the feature for a cloud service within your account. For more information, see Supported Alibaba Cloud services. For example, if Security Center is activated but the **Log Analysis** feature is not enabled in the Security Center console, this issue occurs.

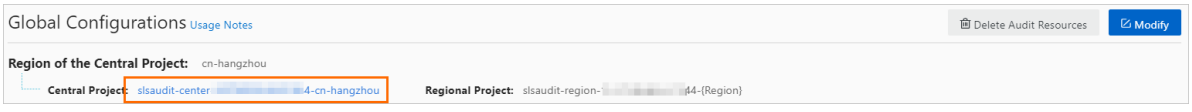
- The number of built-in alert monitoring rules on the **Alert Center** page of the `slsaudit-center-${uid}-${region}` project is different from that on the **Audit Alert** page of Log Audit Service. Why?
  - In addition to the built-in alert monitoring rules for Log Audit Service, the `slsaudit-center-${uid}-${region}` project contains the built-in alert monitoring rules for other features, such as data transformation.
  - The **Audit Alert** page of Log Audit Service displays only the alert monitoring rules for cloud services for which log collection was enabled. This limit does not apply to the **Alert Center** page of the `slsaudit-center-${uid}-${region}` project.
  - The selected region and display language of the Log Service console may also lead to an inconsistency in the number of built-in alert monitoring rules between the **Audit Alert** page of Log Audit Service and the **Alert Center** page of the `slsaudit-center-${uid}-${region}` project.

- The numbers of alert policies, action policies, and alert templates on the **Alert Center** page of the `slsaudit-center-${uid}-${region}` project are different from those on the **Audit Alert** page of Log Audit Service. Why?

  The **Audit Alert** page of Log Audit Service displays only the alert policies, action policies, and alert templates that are related to Log Audit Service. The `slsaudit-center-${uid}-${region}` project may contain the alert policies, action policies, and alert templates that are associated with other applications. Therefore, the inconsistencies arise.

- I cannot find the performance logs of ApsaraDB RDS for MySQL and PolarDB for MySQL instances on the **Audit Query** page. Why?

  The **Audit Query** page displays only the query links for log-type data. Performance logs are of the metric type. You can go to the **Global Configurations** page and click the `slsaudit-center-${uid}-${region}` region. Then, on the **Time Series Storage** page, you can view the performance logs of the ApsaraDB RDS for MySQL and PolarDB for MySQL instances.



- Log Audit Service does not work as expected after I delete the sls-audit-service-monitor role or modify the policy attached to the sls-audit-service-monitor role by mistake in the RAM console. What

do I do?

- If your account is a central account and the sls-audit-service-monitor role created based on your AccessKey pair is used when Log Audit Service is enabled within your account, you can go to the Global Configurations page of Log Audit Service and follow the on-screen instructions to complete the authorization. For more information, see Initially configure Log Audit Service.

  > **Note** Log Audit Service is updated, and the authorization is automatically completed when the AliyunServiceRoleForSLSAudit service-linked role is created. The sls-audit-service-monitor role is still applicable. Make sure that the role exists and its policy is correctly configured. To avoid accidental deletion or illegal modifications, we recommend that you complete the authorization by creating the AliyunServiceRoleForSLSAudit service-linked role.

- If your account is a member, you can modify the policy of the sls-audit-service-monitor role in the RAM console. If the role is not displayed in the RAM console, create the role. For more information, see Use a custom policy to authorize Log Service to collect and synchronize logs.

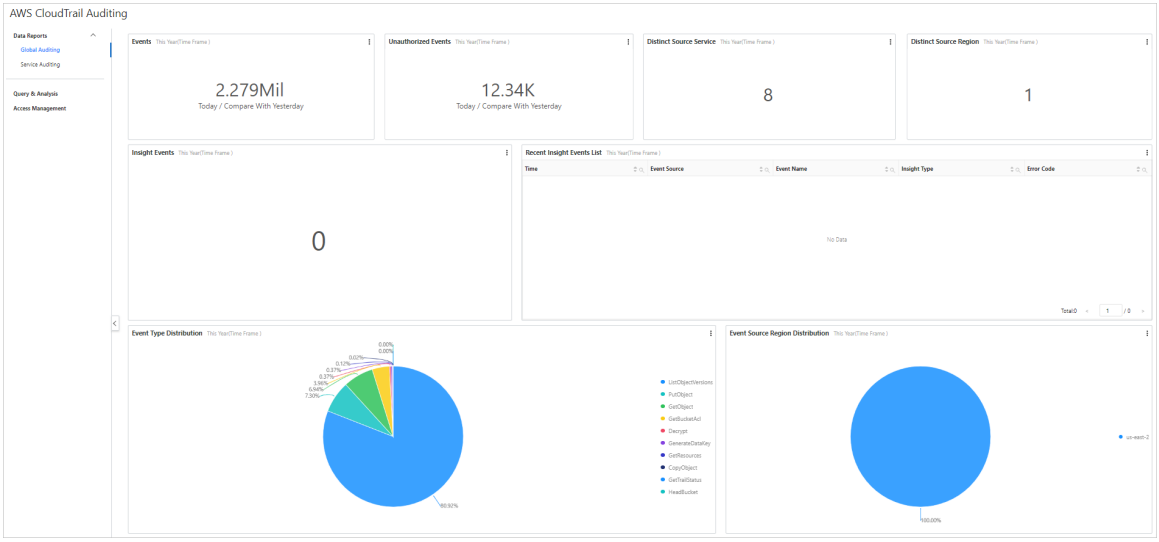
2.AWS CloudTrail Audit

2.1. Usage notes

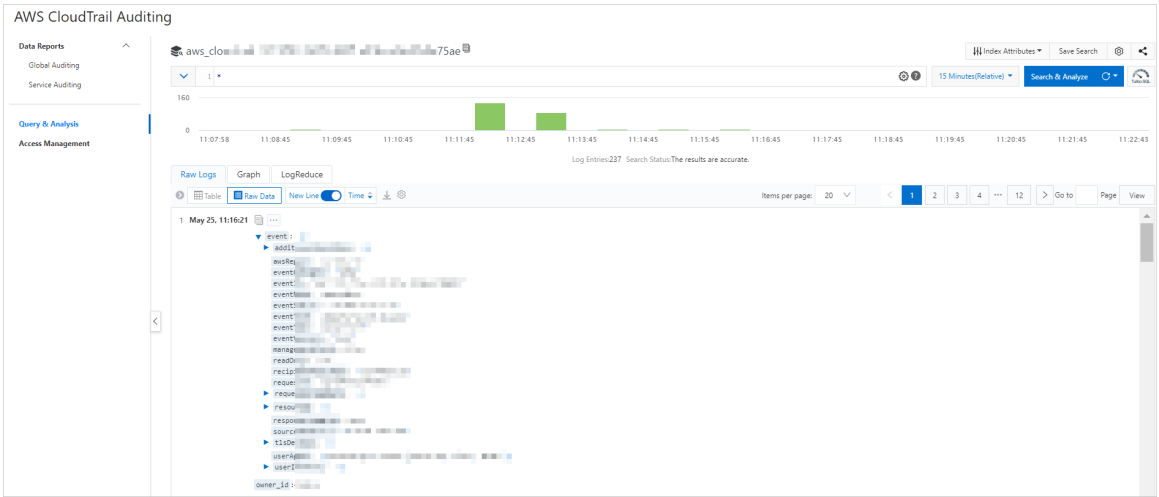
The AWS CloudTrail Audit application of Log Service allows you to pull, store, query, analyze, and visualize Amazon Web Services (AWS) CloudTrail logs. This helps audit the events of your AWS account. This topic describes the features, assets, and billing of the AWS CloudTrail Audit application. This topic also describes how the application works.

Features

- The AWS CloudTrail Audit application provides convenient configurations. This way, you can access AWS CloudTrail data in an efficient manner. For more information, see Create an AWS CloudTrail Audit configuration.
- The AWS CloudTrail Audit application provides out-of-the-box dashboards to help you analyze and audit various events of your AWS account. The dashboards are classified into the following categories: Global Auditing and Service Auditing.

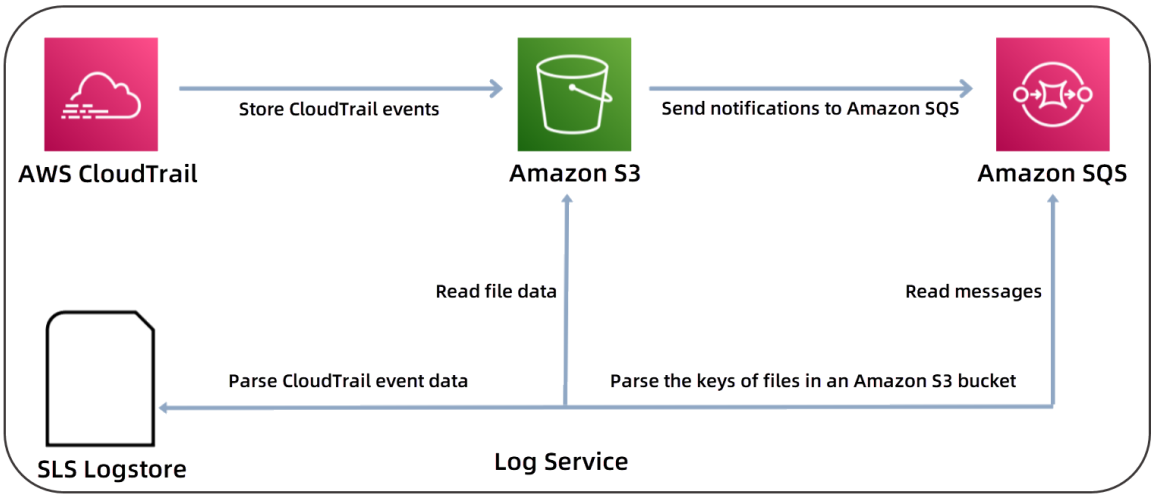


- The AWS CloudTrail Audit application supports custom query and analysis of collected data.



How the AWS CloudTrail Audit application works

To pull AWS CloudTrail data to the AWS CloudTrail Audit application, you must create a trail in the AWS CloudTrail console and create a queue in the Amazon Simple Queue Service (SQS) console.



Assets

You can view the assets of the AWS CloudTrail Audit application in the project that you specify. The following assets are included:

- Logstore

  After you create an AWS CloudTrail Audit configuration, Log Service automatically generates a Logstore named aws_cloudtrail_**** to store AWS CloudTrail data. Log Service also creates indexes for the Logstore.

- Dashboards

Dashboard		Description
Global Auditing	Overview	Displays the overall information of all events that are recorded by AWS CloudTrail in charts. The information includes the number of events, number of source services, number of source regions, number of Insights events, distribution of event types, distribution of source regions, and event trends.
	Logon Auditing	Displays information about the sign-in events that are recorded by AWS CloudTrail in charts. The information includes the distribution of global sign-in events, trends of successful sign-in events and failed sign-in events, distribution of failed authentication events, and global distribution of failed authentication events.

Dashboard		Description
Service Auditing	S3 Data Event	Displays information about Amazon Simple Storage Service (S3) data events that are recorded by AWS CloudTrail in charts. The information includes the list of buckets, number of operations on objects, number of read operations on objects, number of write operations on objects, number of delete operations on objects, and trend of operations on objects. **Note** The dashboard displays data only if AWS CloudTrail that you configure records data events. For more information, see Data events.
	IAM Auditing	Displays information about Identity and Access Management (IAM) events that are recorded by AWS CloudTrail in charts. The information includes the number of error events, distribution of IAM error events, list of error events, distribution of user change events, and list of user change events.
	Network and Security Auditing	Displays information about network and security events that are recorded by AWS CloudTrail in charts. The information includes the distribution of change events for virtual private clouds (VPCs), list of change events for VPCs, distribution of change events for network firewalls, and list of change events for network firewalls.

Billing

- You are charged for the read traffic on Amazon SQS and Amazon S3 buckets. For more information, see AWS pricing.
- After data is stored in Log Service, you are charged for the storage space, read traffic, number of requests, data transformation, and data shipping. For more information, see Billable items.

2.2. Create an AWS CloudTrail Audit configuration

After you complete the required configurations in Amazon Web Services (AWS) and create an AWS CloudTrail Audit configuration in Log Service, you can import AWS CloudTrail data to the AWS CloudTrail Audit application. This topic describes how to create an AWS CloudTrail Audit configuration.

Preparations

Before you can create an AWS CloudTrail Audit configuration, you must complete the following configurations in AWS. The configurations allow Amazon Simple Storage Service (S3) to send notifications to Amazon Simple Queue Service (SQS) after AWS CloudTrail writes data to an Amazon S3 bucket.

1. Create a trail in the AWS CloudTrail console. For more information, see Create a trail.
2. Create a queue in the Amazon SQS console. For more information, see Create a queue.

3. In the Amazon S3 bucket that you specify when you create a trail in Step , configure Amazon S3 event notifications. For more information, see Configure Amazon S3 event notifications.

   When you configure Amazon S3 event notifications, you must set the destination to which Amazon S3 sends event notifications to the queue that you create in Step .

Note If your account is an Identity and Access Management (IAM) account, you must grant the following permissions to the account. For more information, see Create and attach a policy to an IAM user.

```
{
  "Version": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sqs:ReceiveMessage",
        "sqs:SendMessage",
        "sqs:DeleteMessage",
        "sqs:GetQueueAttributes",
        "sqs:ListQueues",
        "s3:GetObject",
        "kms:Decrypt"
      ],
      "Resource": "*"
    }
  ]
}
```

Procedure

1.
2. On the **Audit & Security** tab in the **Log Application** section, click **AWS CloudTrail Audit**.
3. On the **Access Management** page, click **Add**.
4. In the **Create Configuration** panel, create an AWS CloudTrail Audit configuration.

i. Configure the following parameters.

Parameter	Description
Configuration Name	The name of the AWS CloudTrail Audit configuration.
Project	The name of the project to which the assets of the AWS CloudTrail Audit application belong. **Note** Only the projects in the China (Hangzhou), China (Beijing), China (Zhangjiakou), China (Ulanqab), China (Chengdu), and China (Shenzhen) regions are supported.
AWS Account ID	The ID of your AWS account.
AWS AccessKey ID	The AccessKey ID of your AWS account. **Notice** Make sure that your AccessKey pair has the required permissions to access the AWS resources that you want to manage.
AWS Secret AccessKey	The Secret AccessKey of your AWS account.
AWS Region	The region where the Amazon SQS queue resides.
SQS Queue URL	The identifier of the Amazon SQS queue. For more information, see Queue and message identifiers.
SQS BatchSize	The maximum number of messages that Amazon SQS can pull at a time. Valid values: 1 to 10. Default value: 10.
Import Interval	The scheduling interval between data import tasks. Valid values: 1 to 43200. Default value: 3. Unit: minutes.
Concurrent Tasks	The maximum number of concurrent data import tasks. Valid values: 1 to 20. Default value: 1. **Note** If you want to import a large amount of data, we recommend that you set this parameter to a large value.

ii. Click **Preview**.

Note If the preview fails, you must check the parameter settings based on the error messages. You can go to the next step only when **success** is displayed for the preview.

iii. Click **OK**.

What to do next

You can also perform the following operations on the **Access Management** page.

Operation	Description
View audit logs	Click **View Audit Logs** in the Actions column of your configuration. Then, you are navigated to the Logstore in which raw logs are stored. You can view, query, and analyze the raw logs. For more information, see Query and analyze logs.
View reports	Click **View Reports** in the Actions column of your configuration. Then, you are navigated to the dashboard page on which you can view various audit-related dashboards.
Modify the data retention period	Find the data retention period of your configuration and click the  icon to modify the data retention period for the Logstore in which raw logs are stored.
Modify a configuration	Click **Modify** in the Actions column of your configuration. You can modify parameters such as the name of the configuration and the name of the project.
Delete a configuration	If you no longer use the configuration, click **Delete** in the Actions column of your configuration. **Notice** • If you delete a configuration, the data import tasks for the configuration are deleted, but the Logstore that is created for the configuration is retained. • Data that is imported to the Logstore is stored in the Logstore until the data expires. • After you delete a configuration, you cannot restore the configuration. Proceed with caution.

2.3. View data reports

The AWS CloudTrail Audit application provides out-of-the-box dashboards. You can use the dashboards to analyze and audit all types of events in your Amazon Web Services (AWS) account. The dashboards include Overview, Logon Auditing, S3 Data Event, IAM Auditing, and Network and Security Auditing.

Prerequisites

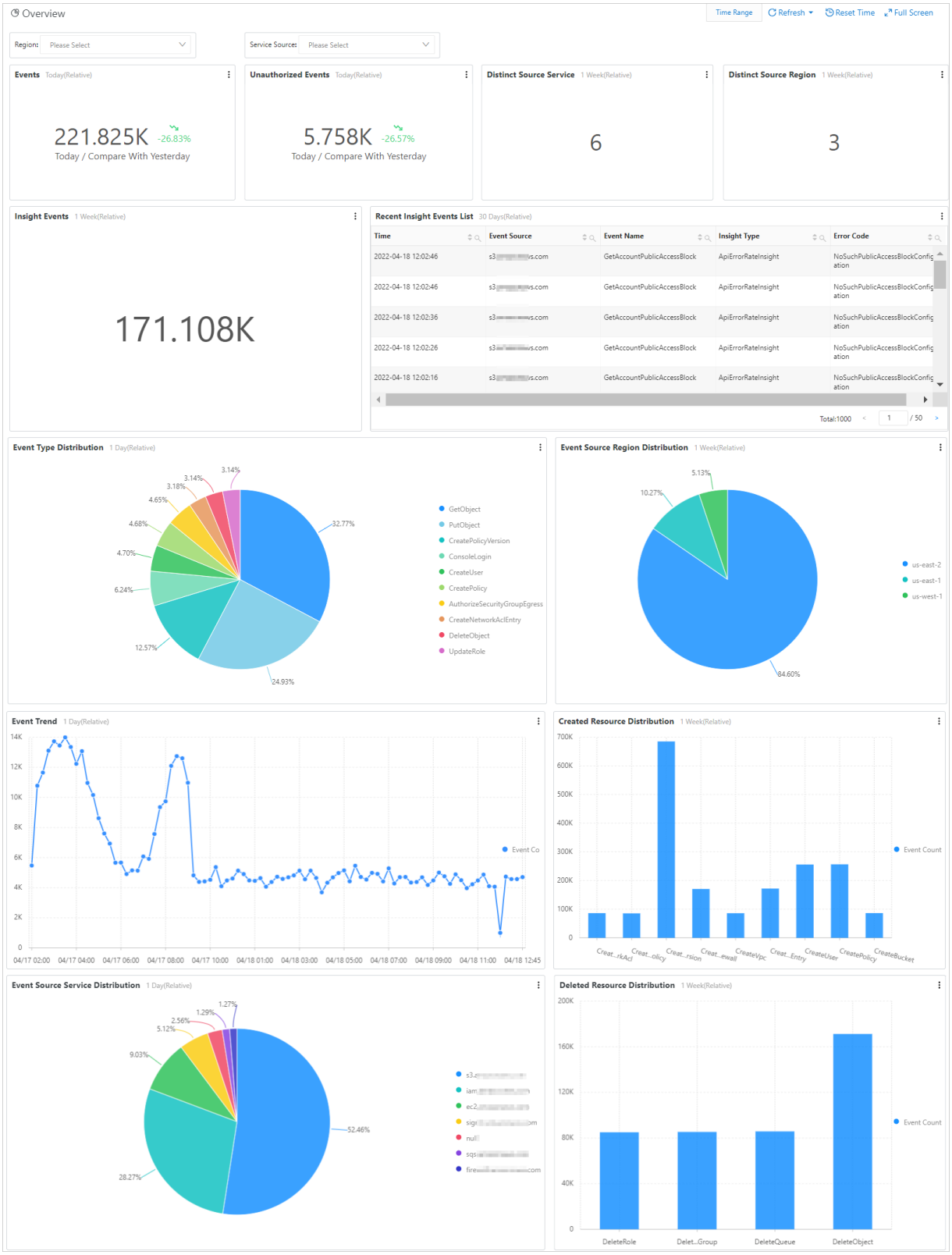
An AWS CloudTrail Audit configuration is created. For more information, see Create an AWS CloudTrail Audit configuration.

Entry point

1.
2. On the **Audit & Security** tab in the **Log Application** section, click **AWS CloudTrail Audit**.
3. In the left-side navigation pane, click the report that you want to view below **Data Reports**.
4. In the upper-left corner of the page that appears, select the AWS CloudTrail Audit configuration.

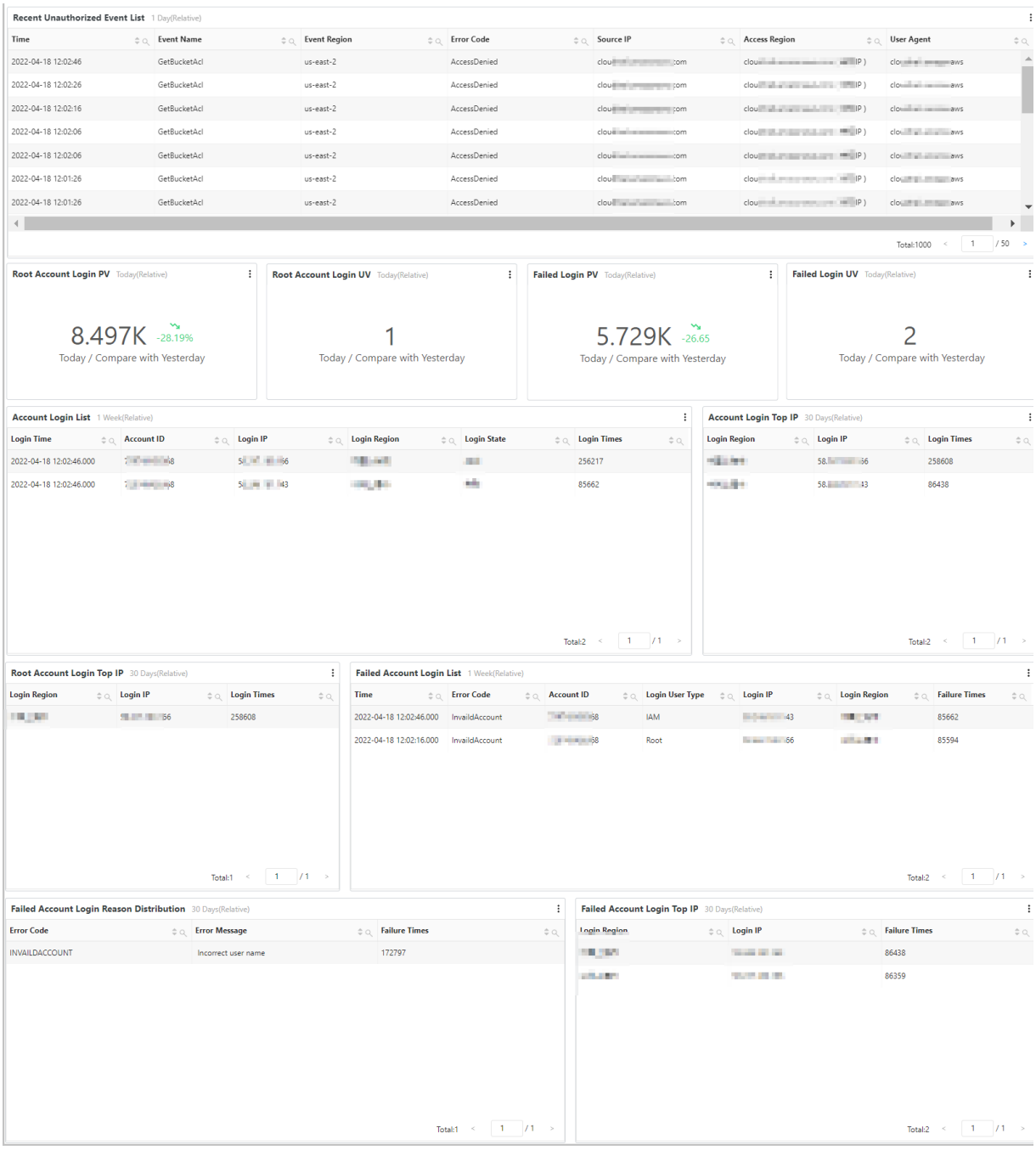
Overview

The **Overview** dashboard displays the overall information of all events that are recorded by AWS CloudTrail in charts. The information includes the number of events, number of source services, number of source regions, number of Insights events, distribution of event types, distribution of source regions, and event trends.



Logon Auditing

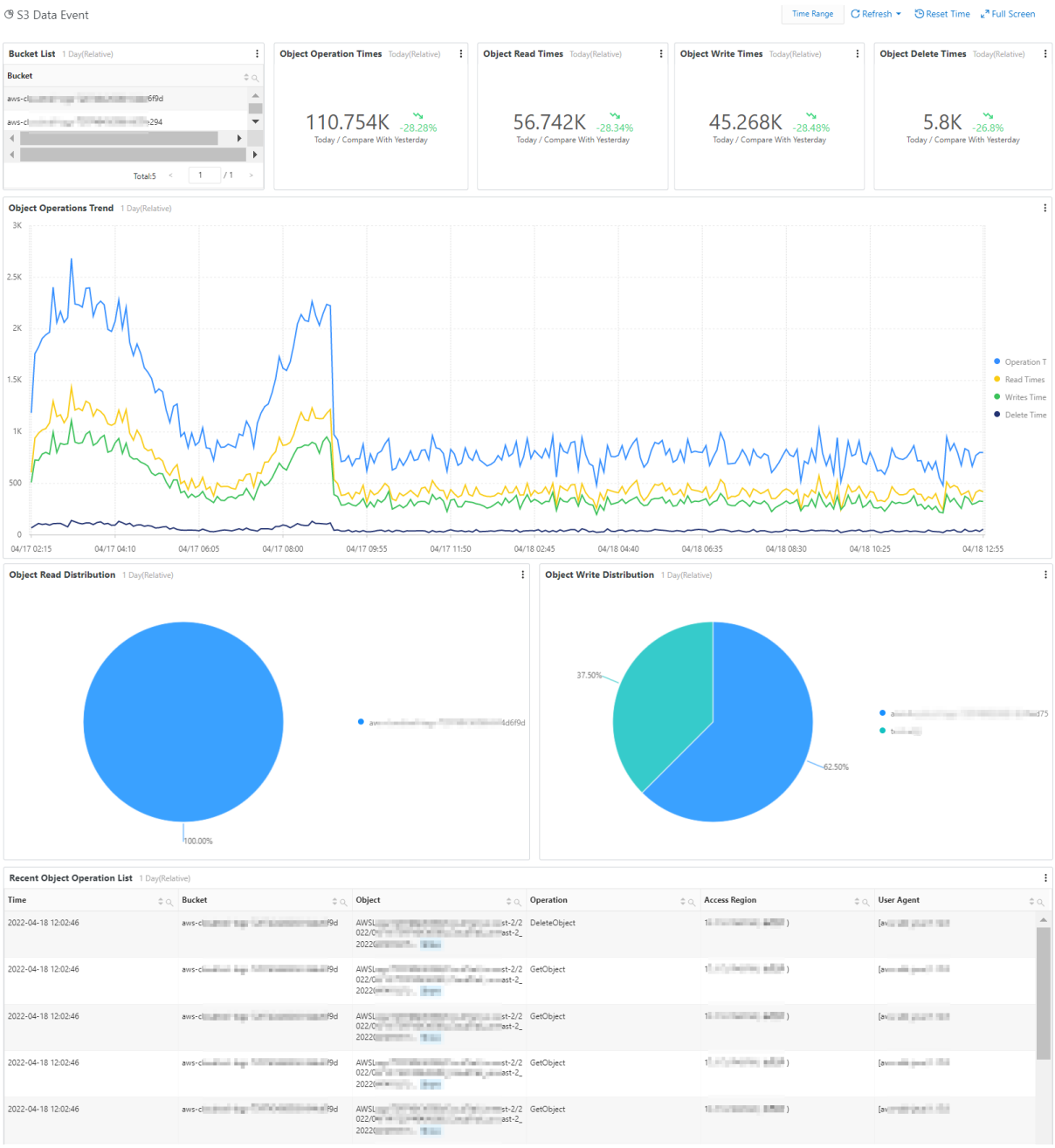
The **Logon Auditing** dashboard displays information about the sign-in events that are recorded by AWS CloudTrail in charts. The information includes the distribution of global sign-in events, trends of successful sign-in events and failed sign-in events, distribution of failed authentication events, and global distribution of failed authentication events.



S3 Data Event

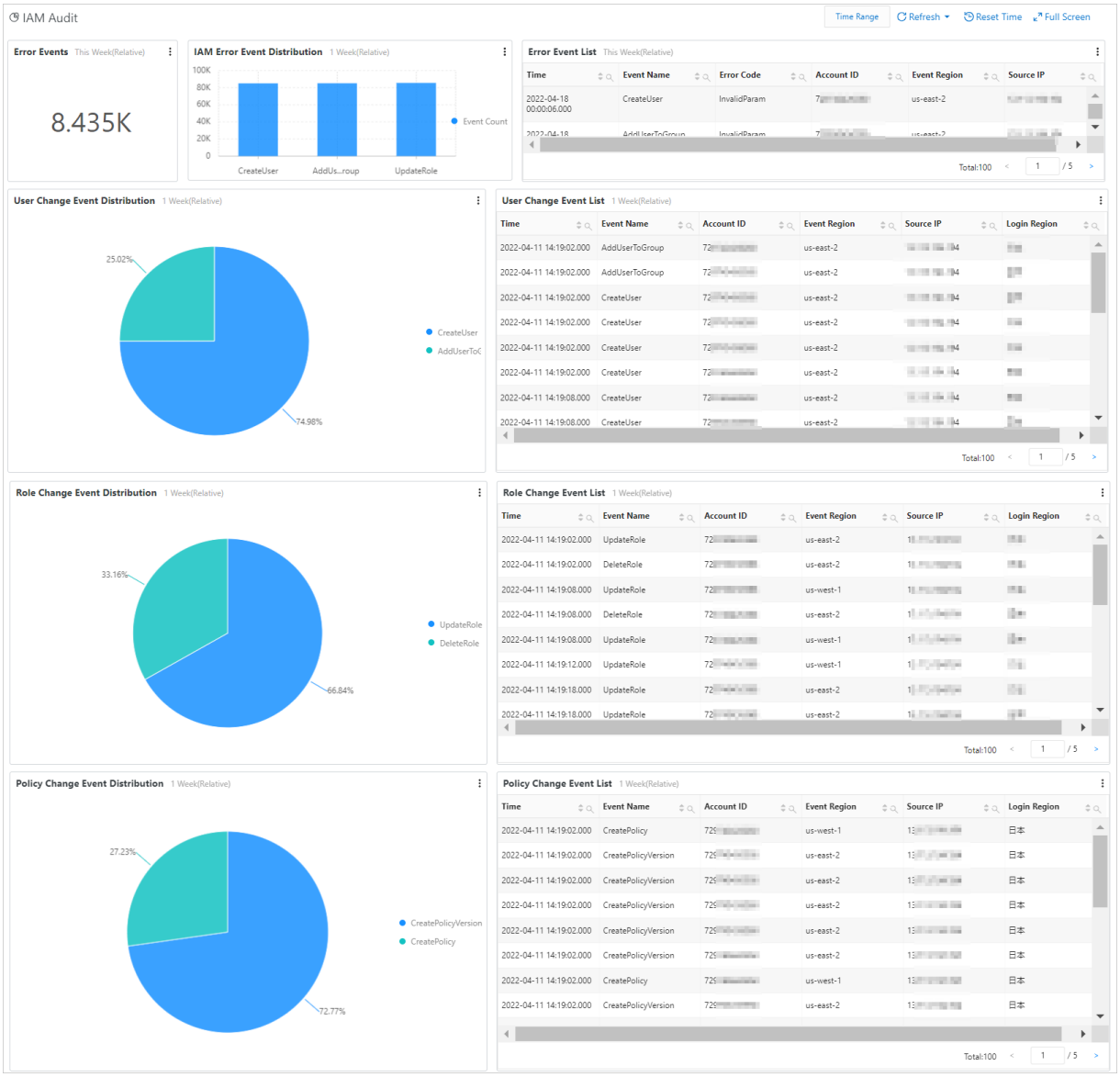
The **S3 Data Event** dashboard displays information about Amazon Simple Storage Service (S3) data events that are recorded by AWS CloudTrail in charts. The information includes the list of buckets, number of operations on objects, number of read operations on objects, number of write operations on objects, number of delete operations on objects, and trend of operations on objects.

> **Note** The dashboard displays data only if AWS CloudTrail that you configure records data events. For more information, see Data events.



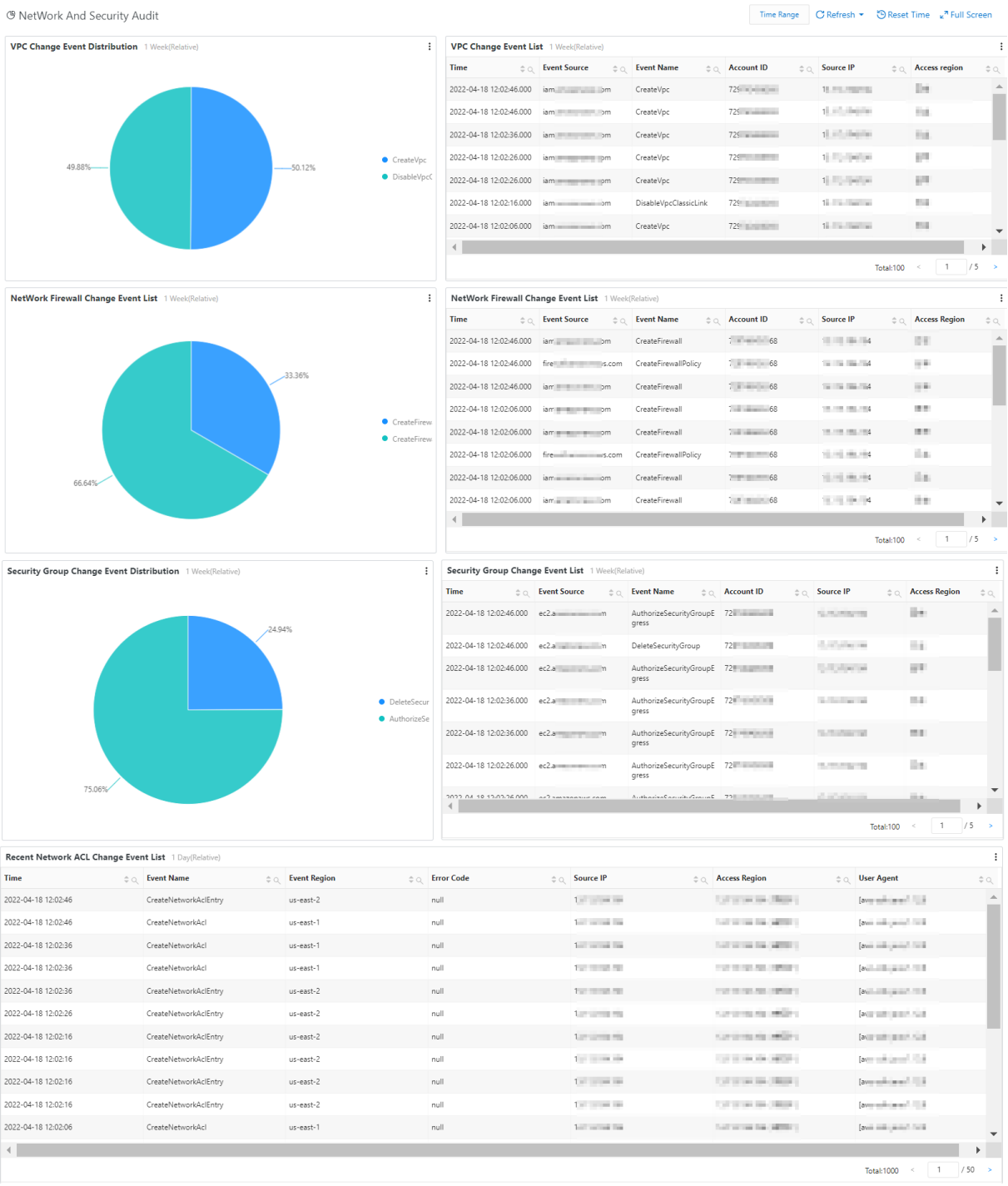
IAM Auditing

The **IAM Auditing** dashboard displays information about Identity and Access Management (IAM) events that are recorded by AWS CloudTrail in charts. The information includes the number of error events, distribution of IAM error events, list of error events, distribution of user change events, and list of user change events.



Network and Security Auditing

The **Network and Security Auditing** dashboard displays information about network and security events that are recorded by AWS CloudTrail in charts. The information includes the distribution of change events for virtual private clouds (VPCs), list of change events for VPCs, distribution of change events for network firewalls, and list of change events for network firewalls.


NetWork And Security Audit
Time Range
Refresh
Reset Time
Full Screen
VPC Change Event Distribution 1 Week(Relative)
49.88%
50.12%
CreateVpc
DisableVpcC
VPC Change Event List 1 Week(Relative)
Time
Event Source
Event Name
Account ID
Source IP
Access region
2022-04-18 12:02:46.000 CreateVpc
2022-04-18 12:02:46.000 CreateVpc
2022-04-18 12:02:36.000 CreateVpc
2022-04-18 12:02:26.000 CreateVpc
2022-04-18 12:02:26.000 CreateVpc
2022-04-18 12:02:16.000 DisableVpcClassicLink
2022-04-18 12:02:06.000 CreateVpc
Total:100
1 / 5
NetWork Firewall Change Event List 1 Week(Relative)
33.36%
66.64%
CreateFirew
CreateFirew
NetWork Firewall Change Event List 1 Week(Relative)
Time
Event Source
Event Name
Account ID
Source IP
Access Region
2022-04-18 12:02:46.000 CreateFirewall
2022-04-18 12:02:46.000 CreateFirewallPolicy
2022-04-18 12:02:46.000 CreateFirewall
2022-04-18 12:02:06.000 CreateFirewall
2022-04-18 12:02:06.000 CreateFirewall
2022-04-18 12:02:06.000 CreateFirewallPolicy
2022-04-18 12:02:06.000 CreateFirewall
2022-04-18 12:02:06.000 CreateFirewall
Total:100
1 / 5
Security Group Change Event Distribution 1 Week(Relative)
24.94%
75.06%
DeleteSecur
AuthorizeSe
Security Group Change Event List 1 Week(Relative)
Time
Event Source
Event Name
Account ID
Source IP
Access Region
2022-04-18 12:02:46.000 AuthorizeSecurityGroupEgress
2022-04-18 12:02:46.000 DeleteSecurityGroup
2022-04-18 12:02:46.000 AuthorizeSecurityGroupEgress
2022-04-18 12:02:36.000 AuthorizeSecurityGroupEgress
2022-04-18 12:02:36.000 AuthorizeSecurityGroupEgress
2022-04-18 12:02:26.000 AuthorizeSecurityGroupEgress
Total:100
1 / 5
Recent Network ACL Change Event List 1 Day(Relative)
Time
Event Name
Event Region
Error Code
Source IP
Access Region
User Agent
2022-04-18 12:02:46 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:46 CreateNetworkAcl us-east-1 null
2022-04-18 12:02:36 CreateNetworkAcl us-east-1 null
2022-04-18 12:02:36 CreateNetworkAcl us-east-1 null
2022-04-18 12:02:36 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:26 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:16 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:16 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:16 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:16 CreateNetworkAclEntry us-east-2 null
2022-04-18 12:02:06 CreateNetworkAcl us-east-1 null
Total:1000
1 / 50

3.Trace

3.1. Usage notes

Log Service provides the distributed tracing feature based on the native OpenTelemetry protocol. You can use this feature to import, store, analyze, visualize trace data, configure alerts for trace data, and manage trace data based on artificial intelligence (AI). This topic describes the background information, import methods, assets, and expenses that are related to the distributed tracing feature (the Trace application) of Log Service.

Background information

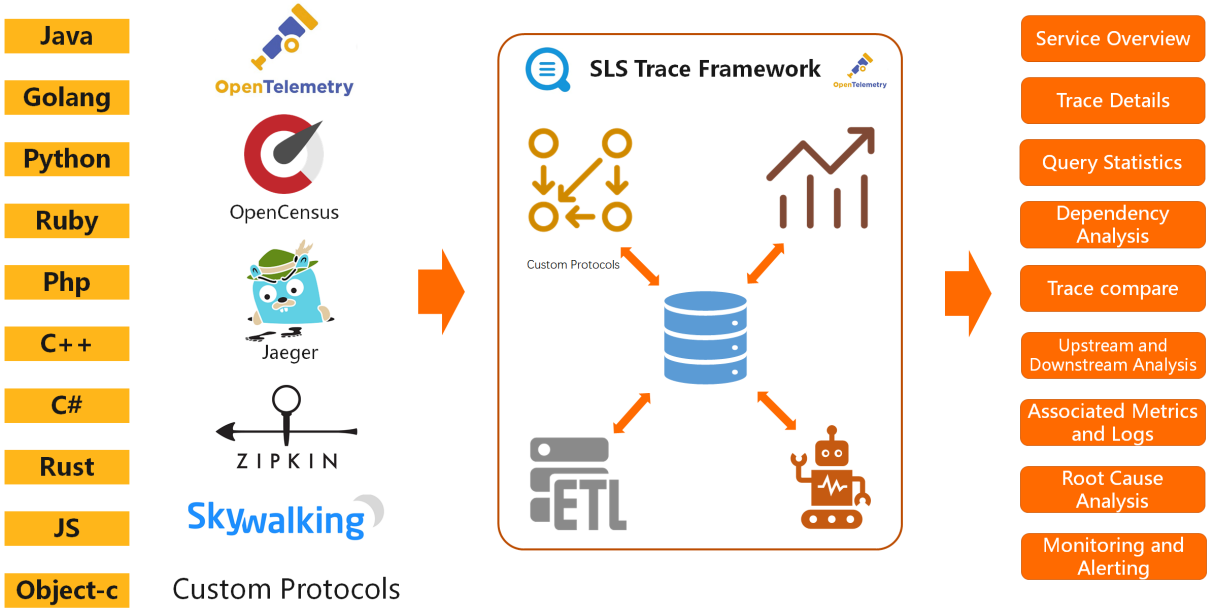
In modern IT systems, especially cloud-native systems and microservice systems, an external request often requires multiple internal services, middleware, and machines to call each other. During the call process, various issues may occur and lead to external service failure or increased latency. This affects user experience. To identify and analyze issues, you can use the distributed tracing feature.

Distributed tracing can provide the call relationships, latency, and results of an entire service call link. This feature is suitable for cloud-native systems, distributed systems, microservices, and other systems that consist of multiple interactive services.

OpenTelemetry is a global standard of distributed tracing, and is compatible with OpenTracing and OpenCensus clients. OpenTelemetry consists of a collection of APIs, SDKs, and tools. You can use OpenTelemetry to instrument, generate, collect, and export various telemetry data, including traces, logs, and metrics.

Trace application of Log Service

OpenTelemetry defines data formats, and generates, collects, and sends data. However, OpenTelemetry does not analyze or visualize data, or configure alerts for data. The Trace application of Log Service is implemented based on the OpenTelemetry protocol. You can use the application to collect trace data from OpenTelemetry and other platforms, such as Jaeger, Zipkin, and SkyWalking. You can also store, analyze, and visualize trace data, and configure alerts for trace data.



- Multiple import methods
  - You can import trace data over multiple protocols such as OpenTelemetry, Jaeger, and Zipkin.
  - You can import trace data in more than 10 programming languages.
  - You can import trace data from multiple trace platforms.
  - You can import trace data over the Internet, Alibaba Cloud internal network, and global acceleration (GA) network. The Alibaba Cloud internal network includes the classic network and virtual private cloud (VPC).
- Complies with the standard specification of OpenTelemetry Trace 1.0

  The trace data format of Log Service complies with OpenTelemetry Trace 1.0 and meets the format requirements for trace data in cloud-native systems and microservices.
- High performance

  You can import petabytes of data per day, extract and analyze metrics, precompute data, and sample 100% of trace data in large-scale scenarios.
- Scalability

  You can customize log storage cycles. The storage capacity of each Logstore can be dynamically scaled to meet your business requirements.
- Various trace features

  You can visualize trace details, view service details, query trace data, analyze dependencies, and customize SQL analysis based on your specific requirements.
- High compatibility with downstream applications

  Log Service trace data and calculated metrics are compatible with various stream processing platforms and offline computing engines. The Trace application also supports customized subscription data.
- Provides multiple built-in AIOps algorithms

  The Trace application can automatically analyze the impact of traces on performance and error rate. This helps developers identify the root causes of various issues in complex scenarios.

Assets

All assets that are created by using the Trace application of Log Service are stored in a specified project. The project consists of the following assets:

- Logstore

  > **Notice** You cannot update or delete indexes in the following Logstores. Otherwise, the Trace application becomes unavailable.

  - {instance}-traces: stores the reported raw trace data.
  - {instance}-traces-metrics: stores the intermediate results of aggregated metrics after trace data is calculated.
  - {instance}-traces-deps: stores the intermediate results of dependencies after trace data is calculated.
  - {instance}-logs: stores the reported raw logs.
- Metricstore

{instance}-metrics: stores the reported metrics.

- Scheduled SQL
  - {instance}-metric_info: queries the metrics that are used to aggregate trace data.
  - {instance}-service: queries the dependencies that are used to aggregate trace data.
  - {instance}-service_name_host: queries the dependencies that are used to aggregate the service, name, and host granularities.
  - {instance}-service_name_host_resource: queries the dependencies that are used to aggregate the service, name, host, resource granularities.
- Dashboard
  - Import overview: displays the basic information of imported trace data, such as the number of traces, number of spans, and import status of each service.
  - Statistics: displays the statistics of imported trace data, such as latency, QPS, and error rate.

Billing

When you use the Trace application, you are charged basic fees for Log Service resources. The basic fees include the index traffic fee, storage fee, read/write traffic fee, and Internet access fee. For more information about billable items, see Billable items.

3.2. Trace data formats

This topic describes the trace data formats that are supported by Log Service.

The trace data formats supported by Log Service are compatible with the data formats defined in OpenTelemetry Trace 1.0 format. The trace data that is written over the OpenTelemetry, Jaeger, Zipkin, OpenCensus, and SkyWalking protocols can be automatically mapped to the trace data formats of OpenTelemetry. Other types of trace data can be transformed to the Log Service Trace format.

Field	Type	Required	Description	Example
host	String	No	The hostname of the host where the resources reside. The host field is extracted from the host.name field in the resource field.	test-host
service	String	Yes	The service name of the resource. The service field is extracted from the service.name field in the resource field.	test-service
resource	JSON Object	No	Resource fields other than host and service, such as process.pid, process.runtime.name, and pod.name. For more information, see Resource Semantic Conventions.	{"k8s.pod.name":"xxxx", "k8s.pod.namespace":"kube-system"}
otlp.name	String	No	The name of the Trace SDK.	go-sdk
otlp.version	String	No	The version of the Trace SDK.	v1.0.0

Field	Type	Required	Description	Example
name	String	Yes	The name of the span.	/get/314159
kind	String	No	The span type, for example, CLIENT and SERVER. For more information, see SpanKind.	SERVER
traceID	String	Yes	The ID of the trace, in hexadecimal.	0123456789abcde0123456789abcde
spanID	String	Yes	The ID of the span, in hexadecimal.	0123456789abcde
parentSpanID	String	Yes	The ID of the parent span, in hexadecimal.	0123456789abcde
links	JSON Array	No	Other associated spans. For more information, see Specifying links.	[{"TraceID" : "abc", "SpanId" : "abc", "TraceState" : "", "Attributes" : { "k" : "v" } }]
logs	JSON Array	No	The associated logs and events. For more information, see Add Events.	None
traceState	String	No	The tracestate header defined by W3C. For more information, see W3C Trace Context Specification.	None
start	INT	Yes	The start time. The value is a UNIX timestamp. Unit: microseconds.	1615882567123456
end	INT	No	The end time. The time is a UNIX timestamp. Unit: microseconds.	1615882567234567
duration	INT	Yes	The latency, which is the difference between the valueof the start parameter and the value of the end parameter. Unit: microseconds.	1020

Field	Type	Required	Description	Example
attribute	JSON Object	Yes	The attribute information of the span, such as the URL and status code of HTTP requests. For more information, see Attribute Naming.	{"custom":"custom","host.hostname":"myhost","my-label":"myapp-type","null-value":"","service.name":"myapp"}
statusCode	String	Yes	The status code. Valid values: OK, ERROR, and UNSET. UNSET and OK are equivalent.	ERROR
statusMessage	String	No	The status message.	stack overflow

3.3. Create a trace instance

A trace instance of Log Service is used to manage all collected trace data. You can query and analyze trace data, and view the details of trace data. You can also view service metrics. This topic describes how to create a trace instance in the Log Service console.

Procedure

1.
2. On the **Intelligent O&M** tab in the **Log Application** section, click **Trace**.
3. Click **Create Instance**.
4. In the **Create Instance** panel, set the parameters and click **OK**. The following table describes the parameters.

Parameter	Description
Name	The name of the trace instance.
Description	The description of the trace instance.
Project	The project that is used to store trace data. If no project is available, click **Create Now** to create a project. For more information, see Create a project.
Instance ID	The ID of the trace instance.
Role Permissions	After trace data is imported, Scheduled SQL jobs are generated in the Trace application to read data from the Logstores that store the trace data. The Scheduled SQL jobs are used to calculate and aggregate metric data. The Scheduled SQL jobs must assume the AliyunLogETLRole system role to read data. If the role is unavailable for the current account, you must complete authorization as prompted.

What's next

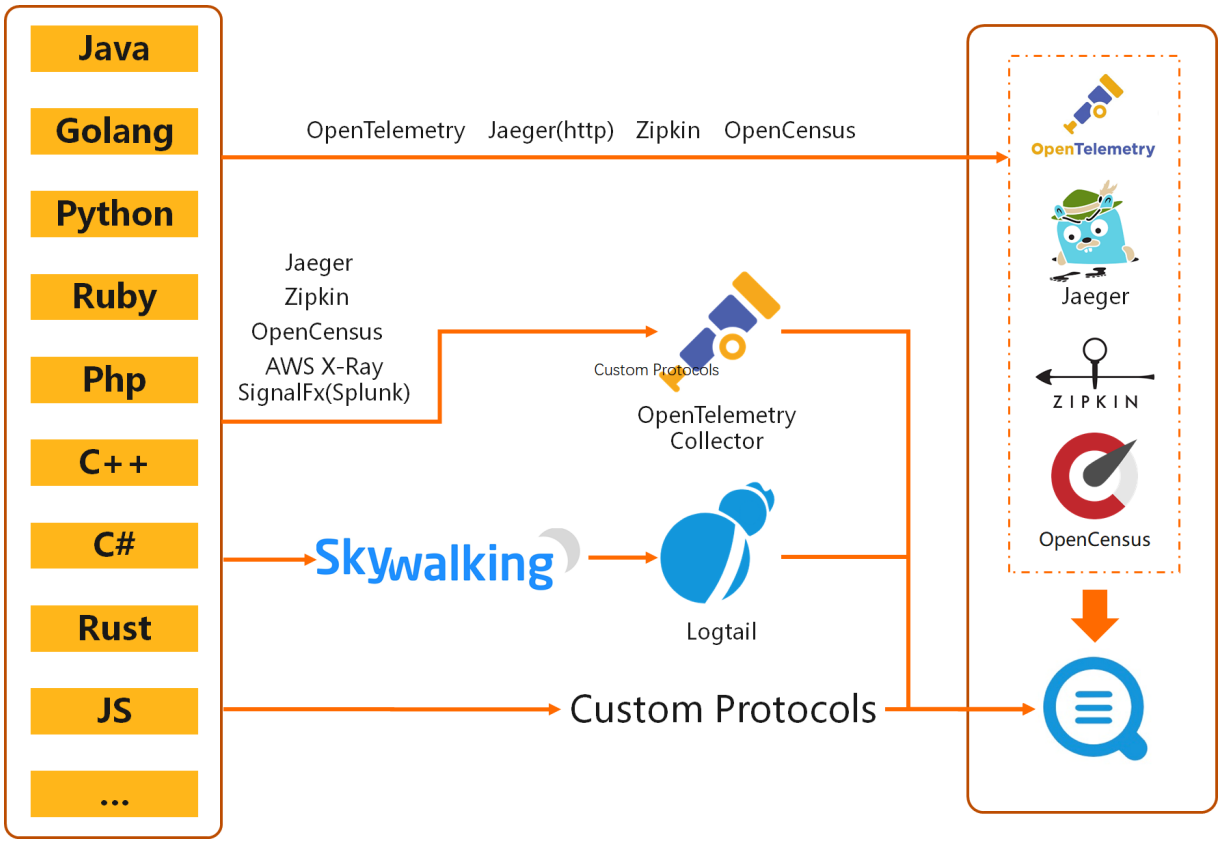
Import trace data

3.4. Import trace data

3.4.1. Overview

You can import cloud native trace data from OpenTelemetry to Log Service. You can also import trace data from other tracing systems to Log Service. This topic describes the methods that can be used to connect to the Trace application of Log Service.

Import methods



Log Service supports the following methods to import trace data:

- Use OpenTelemetry, Jaeger, Zipkin, and OpenCensus to import trace data to Log Service. If you import trace data from Jaeger to Log Service, you can use only an HTTPS or gRPC method.
- Use the OpenTelemetry Collector to forward trace data from OpenTelemetry, Jaeger, Zipkin, OpenCensus, AWS X-Ray, and Splunk SignalFX to Log Service. In this method, all protocols are supported for Jaeger.
- Use Logtail to forward trace data from SkyWalking to Log Service.
- Use a custom protocol to import trace data to Log Service. Then, you can convert the format of the trace data to the OpenTelemetry format by using the data transformation feature of Log Service.

Instructions on selecting import methods

Before you select an import method to import trace data to Log Service, take note of the following instructions:

- Use OpenTelemetry to import trace data.

  The OpenTelemetry protocol is a globally recognized standard that is used to import trace data. To connect with all required components, multiple open source software applications comply with the OpenTelemetry protocol.

- Comply with the OpenTracing or OpenTelemetry protocol to connect with other open source systems.
- If you do not use an open source standard protocol, we recommend that you use the same import method for all services from which trace data is imported in your tracing system. Otherwise, the trace data that is collected may be incomplete.

Details of import methods

Log Service supports multiple import methods that have different automation levels of instrumentation and import complexity. Import methods are listed for common tracing platforms, such as OpenTelemetry, SkyWalking, Jaeger, and Zipkin.

- Import methods for trace data in different programming languages

  You can import trace data to Log Service by using automatic instrumentation or semi-automatic instrumentation.

  - Automatic instrumentation: Developers do not need to modify frameworks or code. Tracing systems automatically set up instrumentation.
  - Semi-automatic instrumentation: Developers need to manually install dependencies or modify code.

Language	Import method	Automation level	Import complexity
Java	Import trace data by using OpenTelemetry	Automatic	Low
	Forward trace data by using the OpenTelemetry Collector	Automatic	Medium
	Import trace data from SkyWalking	Automatic	Medium
Golang	Import trace data by using OpenTelemetry	Semi-automatic	Low
	Forward trace data by using the OpenTelemetry Collector	Semi-automatic	Low
	Import trace data by using OpenTelemetry	Semi-automatic	Medium

Language	Import method	Automation level	Import complexity
Python	Forward trace data by using the OpenTelemetry Collector	Semi-automatic	Medium
NodeJS	Import trace data by using OpenTelemetry	Semi-automatic	Medium
	Forward trace data by using the OpenTelemetry Collector	Semi-automatic	Medium
PHP	Import trace data by using Zipkin	Manual	High
C++	Import trace data by using Jaeger	Manual	High
C#	Import trace data by using OpenTelemetry	Semi-automatic	Medium
	Forward trace data by using the OpenTelemetry Collector	Semi-automatic	Medium
	Import trace data from SkyWalking	Automatic	Medium
Rust	Import trace data by using OpenTelemetry	Manual	High
	Forward trace data by using the OpenTelemetry Collector	Manual	High
Ruby	Import trace data by using OpenTelemetry	Manual	High
	Forward trace data by using the OpenTelemetry Collector	Manual	High

- Import methods for trace data from different platforms

Tracing platform	Import method	Import complexity
	Import trace data from OpenTelemetry	Low

OpenTelemetry Tracing platform	Import method	Import complexity
	Forward trace data by using the OpenTelemetry Collector	Medium
Jaeger	Import trace data from Jaeger	Low
	Forward trace data by using the OpenTelemetry Collector	Medium
Zipkin	Import trace data from Zipkin	Low
	Forward trace data by using the OpenTelemetry Collector	Medium
SkyWalking	Forward trace data by using Logtail	Medium
OpenCensus	Forward trace data by using the OpenTelemetry Collector	Medium
AWS X-Ray	Forward trace data by using the OpenTelemetry Collector	High
Splunk SignalFX	Forward trace data by using the OpenTelemetry Collector	High

Scenarios

- Build a tracing system

  If your system is connected to the Trace application for the first time, we recommend that you use OpenTelemetry to upload your trace data to Log Service. However, the related programming language may not support OpenTelemetry import methods or the import methods may not meet your requirements. In this case, you can use the import methods that support the OpenTracing or OpenCensus protocol to import data from Jaeger or Zipkin.

- Upgrade an existing tracing system

  If your current system uses a tracing service, you can select an import method based on the actual scenario.

  - The tracing system is stably running.
    - If trace data can be uploaded to the OpenTelemetry Collector in the tracing system, you can use the OpenTelemetry Collector to forward the trace data to Log Service.
    - If the tracing system uses a custom protocol or another protocol that is not the OpenTelemetry or OpenTracing protocol, you can print trace data to a file. Then, you can upload the file to Log Service by using Logtail, and use the data transformation feature to convert the data format to the OpenTelemetry format.

- The tracing system does not meet your business requirements or the tracing system needs to be upgraded.
  - If the tracing system uses the OpenTracing or OpenCensus protocol, you can smooth and migrate trace data. In this case, you must upload trace data from the original system to the OpenTelemetry Collector. Then, forward the trace data to Log Service. During this process, the original protocol is replaced by the OpenTelemetry protocol. Then, the OpenTelemetry protocol is used to import the trace data to Log Service.
  - If the tracing system uses another protocol, you must replace the protocol. Otherwise, trace data may be incomplete during the replacement process.

- Deploy on-premises tracing systems

  If you deploy your business applications in a data center and only some gateways connect to the Internet or Express Connect circuits, you can deploy the OpenTelemetry Collector on these gateways. Then, you can send trace data from other machines to the gateways, and then forward the trace data to Log Service by using the OpenTelemetry Collector.

Trace Demo

Log Service provides demos that demonstrate how to import trace data for different programming languages. For more information, see Trace Demos.

3.4.2. New import methods

3.4.2.1. Import trace data from Java applications to Log Service by using OpenTelemetry SDK for Java

This topic describes how to import trace data from Java applications to Log Service by using OpenTelemetry SDK for Java.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Java development environment is set up. The Java version is 8 or later. We recommend that you use Java Development Kit (JDK) 8u252 or a later version.

Method 1: (Recommended) Use a Java agent to automatically upload trace data

You can use a Java agent to automatically upload trace data to Log Service in dozens of Java frameworks. For more information, see Supported libraries, frameworks, application servers, and JVMs.

> **Note** You cannot use a Java agent together with a SkyWalking agent or a Zipkin agent. If you use a Java agent together with a SkyWalking agent or a Zipkin agent, undefined behavior may occur.

1. Download the latest version of Java agent.
2. Configure the Java agent.

The following code provides an example on how to configure the environment variables for the -javaagent parameter of a Java Virtual Machine (JVM). For more information, see opentelemetry-java-instrumentation. You must replace the variables such as *${endpoint}* and *${project}* in the code with the actual values.

```
export OTEL_EXPORTER_OTLP_PROTOCOL=grpc
export OTEL_EXPORTER_OTLP_ENDPOINT=https://${endpoint}
export OTEL_EXPORTER_OTLP_COMPRESSION=gzip
export OTEL_EXPORTER_OTLP_HEADERS=x-sls-otel-project=${project},x-sls-otel-instance-id=
${instance},x-sls-otel-ak-id=${access-key-id},x-sls-otel-ak-secret=${access-key-secret}
java -javaagent:/path/to/opentelemetry-javaagent-all.jar  -Dotel.resource.attributes=se
rvice.namespace=${service.namespace},service.name=${service},service.version=${version}
,host.name=${host},deployment.environment=${environment} -jar /path/to/your/app.jar
```

Variables

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. ○ Port: the port number. The value is fixed as 10010.	test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None

Variable	Description	Example
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None
${service.namespace}	The namespace to which the service belongs.	order
${service}	The name of the service. Specify the value based on your business requirements.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${host}	The hostname.	localhost
${environment}	The deployment environment. Examples: test environment or production environment. Specify the value based on your business requirements.	pre

Method 2: Manually construct and upload trace data

If you use a self-managed framework or have special requirements, you can manually construct trace data and upload the data to Log Service. In this example, Maven is used to construct trace data. For more information, see OpenTelemetry QuickStart.

1. Add Maven dependencies.

```
<dependency>
    <groupId>io.opentelemetry</groupId>
    <artifactId>opentelemetry-sdk</artifactId>
    <version>1.9.0</version>
</dependency>
<dependency>
    <groupId>io.opentelemetry</groupId>
    <artifactId>opentelemetry-exporter-otlp</artifactId>
    <version>1.9.0</version>
</dependency>
<dependency>
    <groupId>io.grpc</groupId>
    <artifactId>grpc-netty-shaded</artifactId>
    <version>1.41.0</version>
</dependency>
<dependency>
    <groupId>io.opentelemetry</groupId>
    <artifactId>opentelemetry-semconv</artifactId>
    <version>1.9.0-alpha</version>
    <scope>runtime</scope>
</dependency>
```

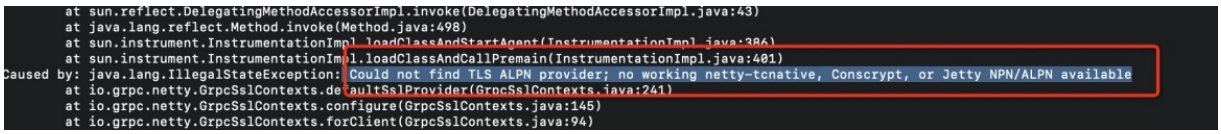
2. Add initialization code.

You must replace the variables such as *${endpoint}* and *${project}* in the following code with the actual values. For more information about the variables, see Variables.

```
OtlpGrpcSpanExporter grpcSpanExporter = OtlpGrpcSpanExporter.builder()
            .setEndpoint("https://${endpoint}")   // You must add https:// to the begin
ning of the value of the .setEndpoint parameter. Example: https://test-project.cn-hangz
hou.log.aliyuncs.com:10010.
            .addHeader("x-sls-otel-project", "${project}")
            .addHeader("x-sls-otel-instance-id", "${instance}")
            .addHeader("x-sls-otel-ak-id", "${access-key-id}")
            .addHeader("x-sls-otel-ak-secret", "${access-key-secret}")
            .build();
        SdkTracerProvider tracerProvider = SdkTracerProvider.builder()
            .addSpanProcessor(BatchSpanProcessor.builder(grpcSpanExporter).build())
            .setResource(Resource.create(Attributes.builder()
                .put(ResourceAttributes.SERVICE_NAME, "${service}")
                .put(ResourceAttributes.SERVICE_NAMESPACE, "${service.namespace}")
                .put(ResourceAttributes.SERVICE_VERSION, "${version}")
                .put(ResourceAttributes.HOST_NAME, "${host}")
                .build()))
            .build();
        OpenTelemetry openTelemetry = OpenTelemetrySdk.builder()
            .setTracerProvider(tracerProvider)
            .setPropagators(ContextPropagators.create(W3CTraceContextPropagator.getInst
ance()))
            .build();
        Tracer tracer =
            openTelemetry.getTracer("instrumentation-library-name", "1.0.0");
        Span parentSpan = tracer.spanBuilder("parent").startSpan();
        try {
            Span childSpan = tracer.spanBuilder("child")
                .setParent(Context.current().with(parentSpan))
                .startSpan();
            childSpan.setAttribute("test", "vllelel");
            // do stuff
            childSpan.end();
        } finally {
            parentSpan.end();
        }
```

FAQ

What do I do if the "Could not find TLS ALPN provider" error message is returned by the Java agent when the OpenJDK version is earlier than 8u252?



To resolve this issue, perform the following steps:

1. Download a package.
2. Run the following command to add the required JAR files.

${youpath} specifies the path to each JAR file. Replace each ${youpath} variable with the actual value.

```
java -Xbootclasspath/p:${youpath}/netty-tcnative-boringssl-static-2.0.25.Final.jar -javaagent:${youpath}/opentelemetry-javaagent-all.jar -jar ${youpath}/demo2-0.0.1-SNAPSHOT.jar
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.2. Import trace data from Golang applications to Log Service by using OpenTelemetry SDK for Golang

This topic describes how to import trace data from Golang applications to Log Service by using OpenTelemetry SDK for Golang.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Golang development environment is set up. The Golang version is 1.13 or later.

Procedure

1. Initialize an OpenTelemetry provider.
2. Check whether the conditions for importing data in semi-automatic mode are met.
   - If the conditions are met, you can import trace data in semi-automatic mode.

     If the semi-automatic mode does not meet your requirements, you must manually import the trace data.
   - If the conditions are not met, you can import trace data in manual mode.

Step 1: Initialize an OpenTelemetry provider

Log Service offers a provider that allows you to build dependencies and upload the dependencies to Log Service. This provider helps simplify the use of an OpenTelemetry provider. For more information, see opentelemetry-go-provider-sls.

> **Notice** You must initialize an OpenTelemetry provider before you create traces and register metrics.

You can run code or configure environment variables to initialize an OpenTelemetry provider.

- Run code to initialize an OpenTelemetry provider.
  i. Add dependencies.

```
module opentelemetry-golang-sample
go 1.13
require (
    github.com/aliyun-sls/opentelemetry-go-provider-sls v0.2.0
    go.opentelemetry.io/contrib/instrumentation/host v0.16.0
    go.opentelemetry.io/contrib/instrumentation/runtime v0.16.0
    go.opentelemetry.io/otel v0.16.0
    go.opentelemetry.io/otel/exporters/otlp v0.16.0
    go.opentelemetry.io/otel/exporters/stdout v0.16.0
    go.opentelemetry.io/otel/sdk v0.16.0
)
```

ii. Write initialization code.

Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
package main
import (
    "github.com/aliyun-sls/opentelemetry-go-provider-sls/provider"
)
func main() {
    slsConfig, err := provider.NewConfig(provider.WithServiceName("${service}"),
        provider.WithServiceNamespace("${service.namespace}"),
        provider.WithServiceVersion("${version}"),
        provider.WithTraceExporterEndpoint("${endpoint}"),
        provider.WithMetricExporterEndpoint("${endpoint}"),
        provider.WithSLSConfig("${project}", "${instance}", "${access-key-id}", "${ac
cess-key-secret}"))
    // Invoke the panic() function. If the initialization fails, the OpenTelemetry pr
ovider exits. You can also use other error handling methods.
    if err != nil {
        panic(err)
    }
    if err := provider.Start(slsConfig); err != nil {
        panic(err)
    }
    defer provider.Shutdown(slsConfig)
    // Add business logic code.
    ...
}${project}
```

Variables

Variable	Description	Example
${service}	The name of the service. Specify the value based on your business requirements.	payment
${service.namespace}	The namespace to which the service belongs.	order
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ■ ${project}: the name of the Log Service project. ■ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. ■ Port: the port number. The value is fixed as 10010. Note ■ If you set the variable to stdout, data is printed to standard output. In this case, the code line is provider.WithTraceExporterEndpoint ("stdout"). ■ If you leave the variable empty, trace data is not uploaded to Log Service.	test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None

- Configure environment variables to initialize an OpenTelemetry provider.

Configuration method	Environment variable	Required	Description	Default value
WithServiceName	SLS_OTEL_SERVICE_NAME	Yes	The name of the service. Specify the value based on your business requirements.	None
WithServiceNAmespace	SLS_OTEL_SERVICE-NAMESPACE	No	The namespace to which the service belongs.	order
WithServiceVersion	SLS_OTEL_SERVICE_VERSION	Yes	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.0
WithSLSConfig	SLS_OTEL_PROJECT, SLS_OTEL_INSTANCE_ID, SLS_OTEL_ACCESS_KEY_ID, and SLS_OTEL_ACCESS_KEY_SECRET	No	The information about Log Service resources. The information includes the name of a project, name of a trace instance, AccessKey ID of an account that has only the write-permissions on the project, and AccessKey secret of the account. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None

Configuration method	Environment variable	Required	Description	Default value
WithTraceExporterEndpoint	SLS_OTEL_TRACE_ENDPOINT	No	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ◦ ${project}: the name of the Log Service project. ◦ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a VPC. A public endpoint can be accessed over the Internet. For more information, see Endpoints. ◦ Port: the port number. The value is fixed as 10010. **Note** ◦ If you set the variable to stdout, data is printed to standard output. ◦ If you leave the variable empty, trace data is not uploaded to Log Service.	stdout

Configuration method	Environment variable	Required	Description	Default value
WithTraceExporterInsecure	SLS_OTEL_TRACE_INSECURE	No	Specifies whether to transfer data by using a method that is not secure. ○ true: transfers data by using a method that is not secure. ○ false: transfers data by using a method that is secure. **Note** If you want to directly transfer data to Log Service, you must set the variable to false.	false
WithMetricExporterEndpoint	SLS_OTEL_METRIC_ENDPOINT	No	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a VPC. A public endpoint can be accessed over the Internet. For more information, see Endpoints. ○ Port: the port number. The value is fixed as 10010. **Note** ○ If you set the variable to stdout, data is printed to standard output. ○ If you leave the variable empty, metric data is not uploaded to Log Service.	stdout

Configuration method	Environment variable	Required	Description	Default value
WithMetricExporterInsecure	SLS_OTEL_METRIC_INSECURE	No	Specifies whether to transfer data by using a method that is not secure. ○ true: transfers data by using a method that is not secure. ○ false: transfers data by using a method that is secure. **Note** If you want to directly transfer data to Log Service, you must set the variable to false.	false
WithResourceAttributes	None	No	The additional tag information, such as the environment and zone.	None
WithResource	OTEL_RESOURCE_ATTRIBUTES	No	The additional tag information, such as the environment and zone. Format: key1=value1,key2=value2.	None
WithMetricReportingPeriod	SLS_OTEL_METRIC_EXPORT_PERIOD	No	The interval of reporting metric data. We recommend that you set the interval to a value from 15s to 60s.	30s
WithErrorHandler	None	No	The error handling function.	None
WithErrorHandlerFunc	None	No	The error handling function.	None
None	SLS_OTEL_ATTRIBUTES_ENV_KEYS	No	The additional tag information, such as the environment and zone. This variable is similar to OTEL_RESOURCE_ATTRIBUTES. However, the values of attribute keys that are defined in the SLS_OTEL_ATTRIBUTES_ENV_KEYS variable are read from other environment variables. SLS_OTEL_ATTRIBUTES_ENV_KEYS is commonly used in Kubernetes clusters to pad some template values to specified environment variables. Format: env-key-1|env-key-2|env-key-3.	None

Step 2: Import data

- Semi-automatic mode: recommended

  OpenTelemetry provides automatic instrumentation solutions for various basic libraries. If your business rely on these libraries, you can use the automatic instrumentation solutions to import data. For more information about basic libraries, see Instrumentation.

  - Use the .NET or HTTP framework to import data

    The following sample code is created based on go.opentelemetry.io/contrib/instrumentation/net/http/otelhttp v0.16.0. For more information, see otel-http-example.

    Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
package main
import (
    "fmt"
    "io"
    "net/http"
    "time"
    "github.com/aliyun-sls/opentelemetry-go-provider-sls/provider"
    "go.opentelemetry.io/contrib/instrumentation/net/http/otelhttp"
    "go.opentelemetry.io/otel"
    "go.opentelemetry.io/otel/label"
    "go.opentelemetry.io/otel/metric"
    "go.opentelemetry.io/otel/trace"
)
func main() {
    slsConfig, err := provider.NewConfig(provider.WithServiceName("${service}"),
        provider.WithServiceNamespace("${service.namespace}"),
        provider.WithServiceVersion("${version}"),
        provider.WithTraceExporterEndpoint("${endpoint}"),
        provider.WithMetricExporterEndpoint("${endpoint}"),
        provider.WithSLSConfig("${project}", "${instance}", "${access-key-id}", "${acce
ss-key-secret}"))
    // Invoke the panic() function. If the initialization fails, the OpenTelemetry prov
ider exits. You can also use other error handling methods.
    if err != nil {
        panic(err)
    }
    if err := provider.Start(slsConfig); err != nil {
        panic(err)
    }
    defer provider.Shutdown(slsConfig)
    // If you want to analyze metric data in the application, you can register the metr
ics.
    labels := []label.KeyValue{
        label.String("label1", "value1"),
    }
    meter := otel.Meter("aliyun.sls")
    sayDavidCount := metric.Must(meter).NewInt64Counter("say_david_count")
    helloHandler := func(w http.ResponseWriter, req *http.Request) {
        if time.Now().Unix()%10 == 0 {
```

```
            _, _ = io.WriteString(w, "Hello, world!\n")
        } else {
            // If you want to record some events, you can obtain the span in the contex
t and add events.
            ctx := req.Context()
            span := trace.SpanFromContext(ctx)
            span.AddEvent("say : Hello, I am david", trace.WithAttributes(label.KeyValu
e{
                Key:   "label-key-1",
                Value: label.StringValue("label-value-1"),
            }))
            _, _ = io.WriteString(w, "Hello, I am david!\n")
            sayDavidCount.Add(req.Context(), 1, labels...)
        }
    }
    // To use the automatic instrumentation solution for otel net/http, you need only t
o enclose http. Handler with otelhttp.NewHandler.
    otelHandler := otelhttp.NewHandler(http.HandlerFunc(helloHandler), "Hello")
    http.Handle("/hello", otelHandler)
    fmt.Println("Now listen port 8080, you can visit 127.0.0.1:8080/hello .")
    err = http.ListenAndServe(":8080", nil)
    if err != nil {
        panic(err)
    }
}
```

- Use the Gorilla Mux framework to import data

  The following sample code is created based on go.opentelemetry.io/contrib/instrumentation/github.com/gorilla/mux/otelmux v0.16.0. The interface may change in later versions. For more information about the latest sample code, see otel-mux-example.

  Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
package main
import (
    "context"
    "fmt"
    "net/http"
    "github.com/aliyun-sls/opentelemetry-go-provider-sls/provider"
    "github.com/gorilla/mux"
    "go.opentelemetry.io/contrib/instrumentation/github.com/gorilla/mux/otelmux"
    "go.opentelemetry.io/otel"
    "go.opentelemetry.io/otel/label"
    "go.opentelemetry.io/otel/metric"
    "go.opentelemetry.io/otel/trace"
)
func main() {
    slsConfig, err := provider.NewConfig(provider.WithServiceName("${service}"),
        provider.WithServiceNamespace("${service.namespace}"),
        provider.WithServiceVersion("${version}"),
        provider.WithTraceExporterEndpoint("${endpoint}"),
        provider.WithMetricExporterEndpoint("${endpoint}"),
```

```
        provider.WithSLSConfig("${project}", "${instance}", "${access-key-id}", "${acce
ss-key-secret}"))
    // Invoke the panic() function. If the initialization fails, the OpenTelemetry prov
ider exits. You can also use other error handling methods.
    if err != nil {
        panic(err)
    }
    if err := provider.Start(slsConfig); err != nil {
        panic(err)
    }
    defer provider.Shutdown(slsConfig)
    // If you want to analyze metric data in the application, you can register the metr
ics.
    labels := []label.KeyValue{
        label.String("label1", "value1"),
    }
    meter := otel.Meter("aliyun.sls")
    callUsersCount := metric.Must(meter).NewInt64Counter("call_users_count")
    r := mux.NewRouter()
    r.Use(otelmux.Middleware("my-server"))
    r.HandleFunc("/users/{id:[0-9]+}", http.HandlerFunc(func(w http.ResponseWriter, r *
http.Request) {
        vars := mux.Vars(r)
        id := vars["id"]
        callUsersCount.Add(r.Context(), 1, labels...)
        name := getUser(r.Context(), id)
        reply := fmt.Sprintf("user %s (id %s)\n", name, id)
        _, _ = w.Write([]byte)(reply))
    }))
    http.Handle("/", r)
    fmt.Println("Now listen port 8080, you can visit 127.0.0.1:8080/users/xxx .")
    _ = http.ListenAndServe(":8080", nil)
}
func getUser(ctx context.Context, id string) string {
    if id == "123" {
        return "otelmux tester"
    }
    // If you want to record some events, you can obtain the span in the context and ad
d events.
    span := trace.SpanFromContext(ctx)
    span.AddEvent("unknown user id : "+id, trace.WithAttributes(label.KeyValue{
        Key:   "label-key-1",
        Value: label.StringValue("label-value-1"),
    }))
    return "unknown"
}
```

- Manual mode

  Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
// Copyright The AliyunSLS Authors
//
// Licensed under the Apache License, Version 2.0 (the "License");
```

```
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
//
//     http://www.apache.org/licenses/LICENSE-2.0
//
// Unless required by applicable law or agreed to in writing, software
// distributed under the License is distributed on an "AS IS" BASIS,
// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
// See the License for the specific language governing permissions and
// limitations under the License.
package main
import (
    "context"
    "errors"
    "fmt"
    "math/rand"
    "time"
    "github.com/aliyun-sls/opentelemetry-go-provider-sls/provider"
    "go.opentelemetry.io/otel"
    "go.opentelemetry.io/otel/codes"
    "go.opentelemetry.io/otel/label"
    "go.opentelemetry.io/otel/metric"
    "go.opentelemetry.io/otel/trace"
)
func main() {
    slsConfig, err := provider.NewConfig(provider.WithServiceName("${service}"),
        provider.WithServiceNamespace("${service.namespace}"),
        provider.WithServiceVersion("${version}"),
        provider.WithTraceExporterEndpoint("${endpoint}"),
        provider.WithMetricExporterEndpoint("${endpoint}"),
        provider.WithSLSConfig("${project}", "${instance}", "${access-key-id}", "${access
-key-secret}"))
    // Invoke the panic() function. If the initialization fails, the OpenTelemetry provid
er exits. You can also use other error handling methods.
    if err != nil {
        panic(err)
    }
    if err := provider.Start(slsConfig); err != nil {
        panic(err)
    }
    defer provider.Shutdown(slsConfig)
    mockTrace()
    mockMetrics()
}
func mockMetrics() {
    // Add labels.
    labels := []label.KeyValue{
        label.String("label1", "value1"),
    }
    meter := otel.Meter("ex.com/basic")
    // The observed value, which is used to obtain a measured value on a regular basis. T
he callback function is invoked once per reporting cycle.
    _ = metric.Must(meter).NewFloat64ValueObserver(
        "randval",
```

```
        func(_ context.Context, result metric.Float64ObserverResult) {
            result.Observe(
                rand.Float64(),
                labels...,
            )
        },
        metric.WithDescription("A random value"),
    )
    temperature := metric.Must(meter).NewFloat64ValueRecorder("temperature")
    interrupts := metric.Must(meter).NewInt64Counter("interrupts")
    ctx := context.Background()
    for {
        temperature.Record(ctx, 100+10*rand.NormFloat64(), labels...)
        interrupts.Add(ctx, int64(rand.Intn(100)), labels...)
        time.Sleep(time.Second * time.Duration(rand.Intn(10)))
    }
}
func mockTrace() {
    tracer := otel.Tracer("ex.com/basic")
    ctx0 := context.Background()
    ctx1, finish1 := tracer.Start(ctx0, "foo")
    defer finish1.End()
    ctx2, finish2 := tracer.Start(ctx1, "bar")
    defer finish2.End()
    ctx3, finish3 := tracer.Start(ctx2, "baz")
    defer finish3.End()
    ctx := ctx3
    getSpan(ctx)
    addAttribute(ctx)
    addEvent(ctx)
    recordException(ctx)
    createChild(ctx, tracer)
}
// example of getting the current span
// Obtain the current span.
func getSpan(ctx context.Context) {
    span := trace.SpanFromContext(ctx)
    fmt.Printf("current span: %v\n", span)
}
// example of adding an attribute to a span
// Add an attribute value to the span.
func addAttribute(ctx context.Context) {
    span := trace.SpanFromContext(ctx)
    span.SetAttributes(label.KeyValue{
        Key:   "label-key-1",
        Value: label.StringValue("label-value-1")})
}
// example of adding an event to a span
// Add an event to the span.
func addEvent(ctx context.Context) {
    span := trace.SpanFromContext(ctx)
    span.AddEvent("event1", trace.WithAttributes(
        label.String("event-attr1", "event-string1"),
        label.Int64("event-attr2", 10)))
}
```

```
}
// example of recording an exception
// Record the result of the span and the error information.
func recordException(ctx context.Context) {
    span := trace.SpanFromContext(ctx)
    span.RecordError(errors.New("exception has occurred"))
    span.SetStatus(codes.Error, "internal error")
}
// example of creating a child span
// Create a child span.
func createChild(ctx context.Context, tracer trace.Tracer) {
    // span := trace.SpanFromContext(ctx)
    _, childSpan := tracer.Start(ctx, "child")
    defer childSpan.End()
    fmt.Printf("child span: %v\n", childSpan)
}
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.3. Import trace data from Python applications to Log Service by using OpenTelemetry SDK for Python

This topic describes how to import trace data from Python applications to Log Service by using OpenTelemetry SDK for Python.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Python development environment is set up. The Python version is 3.7 or later.
- OpenTelemetry SDK for Python is installed in your Python development environment.

  If OpenTelemetry SDK for Python is not installed, you can run the following commands to install the SDK:

```
pip install opentelemetry-api==1.7.1
pip install opentelemetry-sdk==1.7.1
pip install opentelemetry-exporter-otlp==1.7.1
```

Procedure

1. Initialize an OpenTelemetry provider.
2. Check whether the conditions for importing data in semi-automatic mode are met.
   - If the conditions are met, you can import trace data in semi-automatic mode.

     If the semi-automatic mode does not meet your requirements, you must manually import the trace data.
   - If the conditions are not met, you can import trace data in manual mode.

Step 1: Initialize an OpenTelemetry provider

You can initialize an OpenTelemetry provider by using the following code. You must replace the variables in the code with the actual values. For more information about the variables, see Variables.

```
# For Opentelemetry
import socket
from opentelemetry import trace
from opentelemetry.exporter.otlp.trace_exporter import OTLPSpanExporter
from opentelemetry.sdk.resources import Resource
from opentelemetry.sdk.trace import TracerProvider
from opentelemetry.sdk.trace.export import BatchSpanProcessor
from opentelemetry.sdk.trace.export import ConsoleSpanExporter
from opentelemetry.sdk.trace.export import SimpleSpanProcessor
class OpenTelemetrySLSProvider(object):
    def __init__(self, namespace="", service="", version="", endpoint='stdout',
                 project=None, instance=None, access_key_id=None, access_key_secret=None):
        '''
        :param namespace: Your service namespace
        :param service: Your Application Service Name
        :param version: Your Application Version
        :param endpoint: console or https://sls_endpoint:10010
        :param project: SLS project
        :param instance: SLS OTEL InstanceId
        :param access_key_id: Aliyun AccesskeyId
        :param access_key_secret:  Aliyun AccesskeySecret
        '''
        self.sls_otel_endpoint = endpoint
        self.sls_otel_project = project
        self.sls_otel_akid = access_key_id
        self.sls_otel_aksecret = access_key_secret
        self.sls_otel_instanceid = instance
        self.local_mode = False
        if endpoint == "stdout":
            self.local_mode = True
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.name": service,
                "service.namespace": namespace,
                "service.version": version})
        else:
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.namespace": namespace,
                "service.name": service,
                "service.version": version,
                "sls.otel.project": self.sls_otel_project,
                "sls.otel.akid": self.sls_otel_akid,
                "sls.otel.aksecret": self.sls_otel_aksecret,
                "sls.otel.instanceid": self.sls_otel_instanceid
            })
    def initTracer(self):
        trace.set_tracer_provider(TracerProvider(resource=self.resource))
        if self.local_mode:
```

```
            trace.get_tracer_provider().add_span_processor(SimpleSpanProcessor(ConsoleSpanE
xporter()))
        else:
            otlp_exporter = OTLPSpanExporter(endpoint=self.sls_otel_endpoint)
            trace.get_tracer_provider().add_span_processor(BatchSpanProcessor(otlp_exporter
))
# debug mode
#sls_ot_provider = OpenTelemetrySLSProvider(service="example", version="v0.1.0")
# write to sls
sls_ot_provider = OpenTelemetrySLSProvider(namespace="${service.namespace}", service="${ser
vice}", version="${version}",
                                           endpoint='${endpoint}',
                                           project="${project}",
                                           instance="${instance}",
                                           access_key_id="${access-key-id}",
                                           access_key_secret="${access-key-secret}"
                                           )
```

Variables

Variable	Description	Example
${service.namespace}	The namespace to which the service belongs.	order
${service}	The name of the service. Specify the value based on your business requirements.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${endpoint}	The endpoint of the Log Service project. Format: https://${project}.${region-endpoint}:Port. • ${project}: the name of the Log Service project. • ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. • Port: the port number. The value is fixed as 10010. **Note** • If you set the variable to stdout (endpoint='stdout'), data is printed to standard output. • If you leave the variable empty, trace data is not uploaded to Log Service.	https://test-project.cn-hangzhou.log.aliyuncs.com:10010

Variable	Description	Example
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None

Step 2: Import data

You can import trace data to Log Service in semi-automatic or manual mode. OpenTelemetry SDK for Python provides various types of instrumentation packages and supports the semi-automatic instrumentation of common frameworks. If you use one of the instrumentation packages, you must import data in semi-automatic mode. For more information, see Instrumentation packages.

- Semi-automatic instrumentation

  In this example, the flask and requests instrumentation packages are used.

  i. Install the instrumentation packages.

  ```
  pip install opentelemetry-instrumentation-flask
  pip install opentelemetry-instrumentation-requests
  ```

  ii. Run code.

  Replace the variables in the following code with the actual values. For more information, see Variables.

  ```
  # flask-example-1.py
  # for flask
  import flask
  import requests
  # for Opentelemetry instrumentation
  import socket
  from opentelemetry.instrumentation.flask import FlaskInstrumentor
  from opentelemetry.instrumentation.requests import RequestsInstrumentor
  # For Opentelemetry
  from opentelemetry import trace
  from opentelemetry.exporter.otlp.trace_exporter import OTLPSpanExporter
  from opentelemetry.sdk.resources import Resource
  ```

```
from opentelemetry.sdk.trace import TracerProvider
from opentelemetry.sdk.trace.export import BatchSpanProcessor
from opentelemetry.sdk.trace.export import ConsoleSpanExporter
from opentelemetry.sdk.trace.export import SimpleSpanProcessor
class OpenTelemetrySLSProvider(object):
    def __init__(self, namespace="", service="", version="", endpoint='stdout',
                 project=None, instance=None, access_key_id=None, access_key_secret=N
one):
        '''
        :param namespace: Your service namespace
        :param service: Your Application Service Name
        :param version: Your Application Version
        :param endpoint: console or https://sls_endpoint:10010
        :param project: SLS project
        :param instance: SLS OTEL InstanceId
        :param access_key_id: Aliyun AccesskeyId
        :param access_key_secret:  Aliyun AccesskeySecret
        '''
        self.sls_otel_endpoint = endpoint
        self.sls_otel_project = project
        self.sls_otel_akid = access_key_id
        self.sls_otel_aksecret = access_key_secret
        self.sls_otel_instanceid = instance
        self.local_mode = False
        if endpoint == "stdout":
            self.local_mode = True
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.name": service,
                "service.namespace": namespace,
                "service.version": version})
        else:
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.name": service,
                "service.version": version,
                "service.namespace": namespace,
                "sls.otel.project": self.sls_otel_project,
                "sls.otel.akid": self.sls_otel_akid,
                "sls.otel.aksecret": self.sls_otel_aksecret,
                "sls.otel.instanceid": self.sls_otel_instanceid
            })
    def initTracer(self):
        trace.set_tracer_provider(TracerProvider(resource=self.resource))
        if self.local_mode:
            trace.get_tracer_provider().add_span_processor(SimpleSpanProcessor(Consol
eSpanExporter()))
        else:
            otlp_exporter = OTLPSpanExporter(endpoint=self.sls_otel_endpoint)
            trace.get_tracer_provider().add_span_processor(BatchSpanProcessor(otlp_ex
porter))
# write to sls
sls_ot_provider = OpenTelemetrySLSProvider(namespace="${service.namespace}", service=
"${service}", version="${version}",
```

```
                                        endpoint='${endpoint}',
                                        project="${project}",
                                        instance="${instance}",
                                        access_key_id="${access-key-id}",
                                        access_key_secret="${access-key-secret}"
                                        )
# for console debug
#sls_ot_provider = OpenTelemetrySLSProvider(service="example", version="v0.1.0")
sls_ot_provider.initTracer()
# flask init
app = flask.Flask(__name__)
# instrumentation init
FlaskInstrumentor().instrument_app(app)
RequestsInstrumentor().instrument()
@app.route("/")
def hello():
    tracer = trace.get_tracer(__name__)
    with tracer.start_as_current_span("request_server"):
        requests.get("http://www.taobao.com")
    return "hello"
app.run(debug=True, port=5000)
```

iii. Access the service to trigger trace data generation and send the trace data to Log Service.

```
127.0.0.1:5000/hello
```

- Manual instrumentation

  Run the following code. Replace the variables in the code with the actual values. For more information about the code, see Variables.

```
# ot-manual-example.py
import time
# For Opentelemetry
import socket
from opentelemetry import trace
from opentelemetry.exporter.otlp.trace_exporter import OTLPSpanExporter
from opentelemetry.sdk.resources import Resource
from opentelemetry.sdk.trace import TracerProvider
from opentelemetry.sdk.trace.export import BatchSpanProcessor
from opentelemetry.sdk.trace.export import ConsoleSpanExporter
from opentelemetry.sdk.trace.export import SimpleSpanProcessor
class OpenTelemetrySLSProvider(object):
    def __init__(self, namespace="", service="", version="", endpoint='stdout',
                 project=None, instance=None, access_key_id=None, access_key_secret=None)
:
        '''
        :param service: Your service namespace
        :param service: Your Application Service Name
        :param version: Your Application Version
        :param endpoint: console or https://sls_endpoint:10010
        :param project: SLS project
        :param instance: SLS OTEL InstanceId
        :param access_key_id: Aliyun AccesskeyId
        :param access_key_secret:  Aliyun AccesskeySecret
```

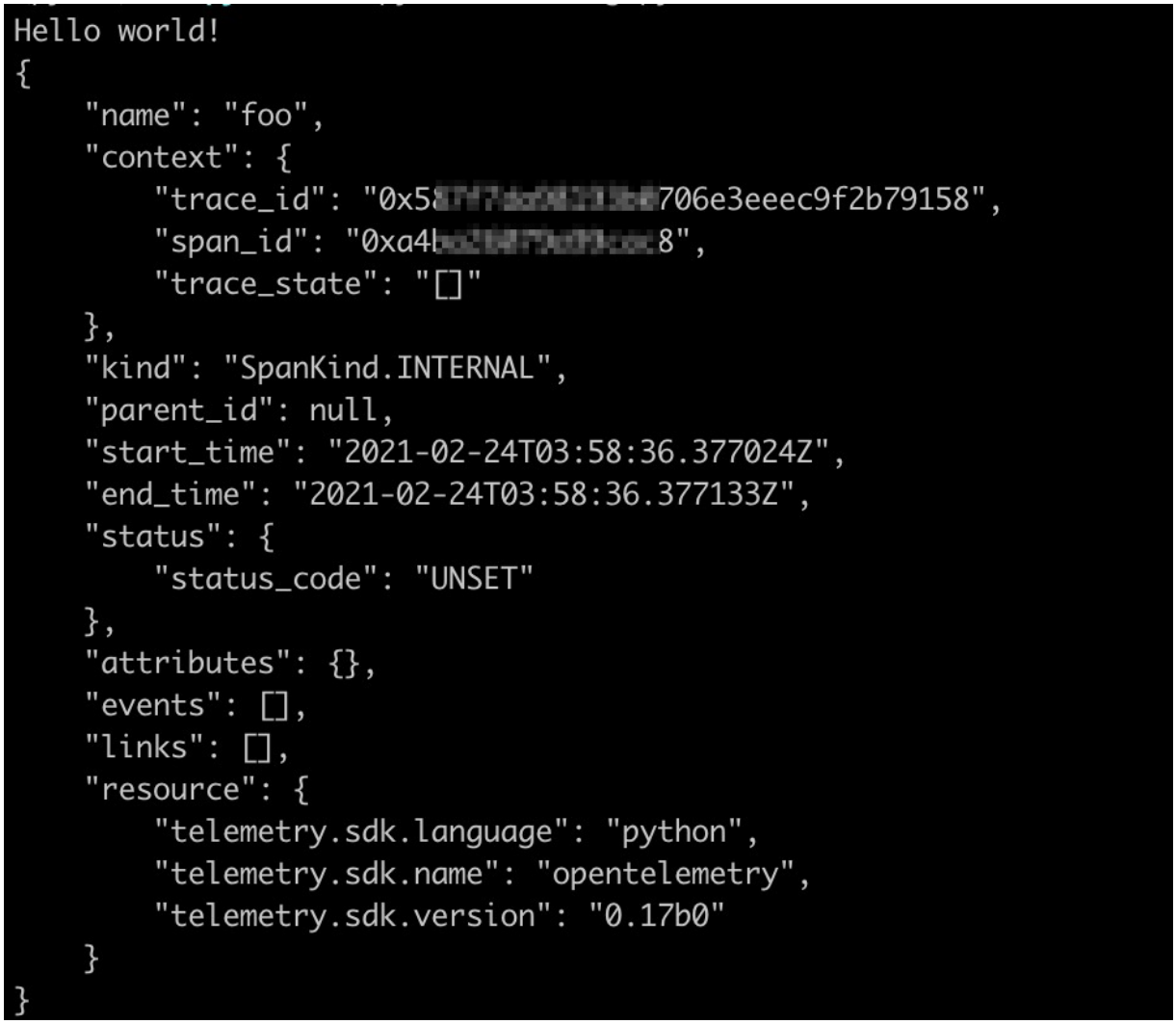
```
        '''
        self.sls_otel_endpoint = endpoint
        self.sls_otel_project = project
        self.sls_otel_akid = access_key_id
        self.sls_otel_aksecret = access_key_secret
        self.sls_otel_instanceid = instance
        self.local_mode = False
        if endpoint == "stdout":
            self.local_mode = True
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.name": service,
                "service.namespace": namespace,
                "service.version": version})
        else:
            self.resource = Resource(attributes={
                "host.name": socket.gethostname(),
                "service.name": service,
                "service.version": version,
                "service.namespace": namespace,
                "sls.otel.project": self.sls_otel_project,
                "sls.otel.akid": self.sls_otel_akid,
                "sls.otel.aksecret": self.sls_otel_aksecret,
                "sls.otel.instanceid": self.sls_otel_instanceid
            })
    def initTracer(self):
        trace.set_tracer_provider(TracerProvider(resource=self.resource))
        if self.local_mode:
            trace.get_tracer_provider().add_span_processor(SimpleSpanProcessor(ConsoleSpa
nExporter()))
        else:
            otlp_exporter = OTLPSpanExporter(endpoint=self.sls_otel_endpoint)
            trace.get_tracer_provider().add_span_processor(BatchSpanProcessor(otlp_export
er))
# write to sls
sls_ot_provider = OpenTelemetrySLSProvider(namespace="${service.namespace}", service="${s
ervice}", version="${version}",
                                           endpoint='${endpoint}',
                                           project="${project}",
                                           instance="${instance}",
                                           access_key_id="${access-key-id}",
                                           access_key_secret="${access-key-secret}"
                                           )
# for console debug
#sls_ot_provider = OpenTelemetrySLSProvider(service="example", version="v0.1.0")
# Trace Example
sls_ot_provider.initTracer()
tracer = trace.get_tracer(__name__)
with tracer.start_as_current_span("foo"):
    print("Hello world!")
labels = {"environment": "staging"}
requests_counter.add(25, labels)
time.sleep(60)
```

FAQ

How do I check whether OpenTelemetry SDK for Python is installed as expected?

The following sample code provides an example on how to check whether the SDK is installed as expected. Save the code as a *tracing.py* file. Then, run the **tracing.py** command. If the command output is returned as expected, the related dependencies of OpenTelemetry SDK for Python are installed.

```
# tracing-example-1.py
from opentelemetry import trace
from opentelemetry.sdk.trace import TracerProvider
from opentelemetry.sdk.trace.export import (
    ConsoleSpanExporter,
    SimpleSpanProcessor,
)
trace.set_tracer_provider(TracerProvider())
trace.get_tracer_provider().add_span_processor(
    SimpleSpanProcessor(ConsoleSpanExporter())
)
tracer = trace.get_tracer(__name__)
with tracer.start_as_current_span("foo"):
    print("Hello world!")
```



What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.4. Import trace data from Node.js applications to Log Service by using OpenTelemetry SDK for JavaScript

This topic describes how to import trace data from Node.js applications to Log Service by using OpenTelemetry SDK for JavaScript.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Node.js development environment is set up. The Node.js version is 8.5.0 or later.

Method 1: (Recommended) Import data in semi-automatic mode

You can install Node.js dependencies in some frameworks to automatically upload trace data to Log Service. The frameworks include HTTP, HTTPS, gRPC, Express, MySQL, MongoDB, and Redis. For more information about the frameworks, see opentelemetry-node-js-contrib. In this example, Express is used to describe how to import data in semi-automatic mode. For more information, see Examples.

1. Install dependencies.

```
npm install --save @opentelemetry/api
npm install --save @opentelemetry/node
npm install --save @opentelemetry/tracing
npm install --save @opentelemetry/exporter-collector-grpc
npm install --save @opentelemetry/instrumentation
npm install --save @opentelemetry/instrumentation-express
npm install --save @opentelemetry/instrumentation-http
npm install --save @grpc/grpc-js
npm install --save @opentelemetry/sdk-trace-node
```

2. Initialize a tracer and start Express.

   Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
const opentelemetry = require("@opentelemetry/api");
const { registerInstrumentations } = require("@opentelemetry/instrumentation");
const { NodeTracerProvider } = require("@opentelemetry/sdk-trace-node");
const { Resource } = require("@opentelemetry/resources");
const {
  SemanticResourceAttributes,
} = require("@opentelemetry/semantic-conventions");
const {
  SimpleSpanProcessor,
  ConsoleSpanExporter,
} = require("@opentelemetry/tracing");
const grpc = require("@grpc/grpc-js");
const {
  CollectorTraceExporter,
} = require("@opentelemetry/exporter-collector-grpc");
const {
  ExpressInstrumentation,
} = require("@opentelemetry/instrumentation-express");
const { HttpInstrumentation } = require("@opentelemetry/instrumentation-http");
var os = require("os");
var hostname = os.hostname();
const provider = new NodeTracerProvider({
  resource: new Resource({
    [SemanticResourceAttributes.SERVICE_NAME]: "${service}",
      [SemanticResourceAttributes.DEPLOYMENT_ENVIRONMENT]: "${environment}",
    [SemanticResourceAttributes.SERVICE_VERSION]: "${version}",
    [SemanticResourceAttributes.SERVICE_NAMESPACE]: "${service.namespace}",
    [SemanticResourceAttributes.HOST_NAME]: hostname,
  }),
});
provider.register();
registerInstrumentations({
  instrumentations: [
    new HttpInstrumentation()
```

```
    new HttpInstrumentation(),
    new ExpressInstrumentation({
      ignoreLayersType: [new RegExp("middleware.*")],
    }),
  ],
  tracerProvider: provider,
});
var url = "${endpoint}";
var logStdout = false;
if (url == "stdout") {
  logStdout = true;
}
var meta = new grpc.Metadata();
meta.add("x-sls-otel-project", "${project}");
meta.add("x-sls-otel-instance-id", "${instance}");
meta.add("x-sls-otel-ak-id", "${access-key-id}");
meta.add("x-sls-otel-ak-secret", "${access-key-secret}");
const collectorOptions = {
  url: url,
  credentials: grpc.credentials.createSsl(),
  metadata: meta,
};
const exporter = new CollectorTraceExporter(collectorOptions);
if (!logStdout) {
  provider.addSpanProcessor(new SimpleSpanProcessor(exporter));
} else {
  var stdexporter = new ConsoleSpanExporter();
  provider.addSpanProcessor(new SimpleSpanProcessor(stdexporter));
}
provider.register();
var tracer = opentelemetry.trace.getTracer("${service}");
var express = require("express");
var app = express();
app.get("/hello", function (req, res, next) {
  res.send("success");
});
var server = app.listen(8079, function () {
  var port = server.address().port;
  console.log("App now running in %s mode on port %d", app.get("env"), port);
});
```

Variables

Variable	Description	Example

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. ○ Port: the port number. The value is fixed as 10010.	test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None
${service}	The name of the service. Specify the value based on your business requirements.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${service.namespace}	The namespace to which the service belongs.	order
${environment}	The deployment environment of the service. Examples: test environment, staging environment, or production environment.	pre

3. Access the service to trigger trace data generation and send the trace data to Log Service.

```
127.0.0.1:8079/hello
```

Method 2: Manually generate and send trace data

If you use a self-managed framework or have special requirements, you can manually construct trace data and send the data to Log Service. For more information, see opentelemetry-js.

1. Install dependencies.

```
npm install --save @opentelemetry/api
npm install --save @opentelemetry/node
npm install --save @opentelemetry/tracing
npm install --save @opentelemetry/exporter-collector-grpc
```

2. Initialize a tracer and start Express.

   Replace the variables in the following code with the actual values. For more information, see Variables.

```
const opentelemetry = require("@opentelemetry/api");
const { registerInstrumentations } = require("@opentelemetry/instrumentation");
const { NodeTracerProvider } = require("@opentelemetry/sdk-trace-node");
const { Resource } = require("@opentelemetry/resources");
const {
  SemanticResourceAttributes,
} = require("@opentelemetry/semantic-conventions");
const {
  SimpleSpanProcessor,
  ConsoleSpanExporter,
} = require("@opentelemetry/tracing");
const grpc = require("@grpc/grpc-js");
const {
  CollectorTraceExporter,
} = require("@opentelemetry/exporter-collector-grpc");
const {
  ExpressInstrumentation,
} = require("@opentelemetry/instrumentation-express");
const { HttpInstrumentation } = require("@opentelemetry/instrumentation-http");
var os = require("os");
var hostname = os.hostname();
const provider = new NodeTracerProvider({
  resource: new Resource({
    [SemanticResourceAttributes.SERVICE_NAME]: "${service}",
      [SemanticResourceAttributes.DEPLOYMENT_ENVIRONMENT]: "${environment}",
    [SemanticResourceAttributes.SERVICE_VERSION]: "${version}",
    [SemanticResourceAttributes.SERVICE_NAMESPACE]: "${service.namespace}",
    [SemanticResourceAttributes.HOST_NAME]: hostname,
  }),
});
provider.register();
registerInstrumentations({
  instrumentations: [
    new HttpInstrumentation(),
    new ExpressInstrumentation({
```

```
      ignoreLayersType: [new RegExp("middleware.*")],
    }),
  ],
  tracerProvider: provider,
});
var url = "${endpoint}";
var logStdout = false;
if (url == "stdout") {
  logStdout = true;
}
var meta = new grpc.Metadata();
meta.add("x-sls-otel-project", "${project}");
meta.add("x-sls-otel-instance-id", "${instance}");
meta.add("x-sls-otel-ak-id", "${access-key-id}");
meta.add("x-sls-otel-ak-secret", "${access-key-secret}");
const collectorOptions = {
  url: url,
  credentials: grpc.credentials.createSsl(),
  metadata: meta,
};
const exporter = new CollectorTraceExporter(collectorOptions);
if (!logStdout) {
  provider.addSpanProcessor(new SimpleSpanProcessor(exporter));
} else {
  var stdexporter = new ConsoleSpanExporter();
  provider.addSpanProcessor(new SimpleSpanProcessor(stdexporter));
}
provider.register();
var tracer = opentelemetry.trace.getTracer("${service}");
var express = require('express');
var app = express()
app.get('/hello', function (req, res, next) {
    const span = tracer.startSpan('hello');
    span.setAttribute('name', 'toma');
    span.setAttribute('age', '26');
    span.addEvent('invoking doWork');
    res.send("success");
    span.end();
});
var server = app.listen(8079, function () {
  var port = server.address().port;
  console.log("App now running in %s mode on port %d", app.get("env"), port);
});
```

3. Access the service to trigger trace data generation and send the trace data to Log Service.

```
127.0.0.1:8079/hello
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.5. Import trace data from C# applications to Log Service by using OpenTelemetry SDK for .NET

This topic describes how to import trace data from C# applications to Log Service by using OpenTelemetry SDK for .NET.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A .NET Framework development environment is set up.

> **Note** The import method in this topic supports all official versions of .NET Framework except for .NET Framework 3.5 SP1. For more information, see .NET Core and .NET Framework.

Procedure

1. Add dependencies.

```
dotnet add package OpenTelemetry --version 1.2.0-beta1
dotnet add package OpenTelemetry.Exporter.Console --version 1.2.0-beta1
dotnet add package OpenTelemetry.Exporter.OpenTelemetryProtocol --version 1.2.0-beta1
dotnet add package OpenTelemetry.Extensions.Hosting --version 1.0.0-rc8
dotnet add package OpenTelemetry.Instrumentation.AspNetCore --version 1.0.0-rc8
dotnet add package Grpc.Core --version 2.36.4
```

2. Run code.

   Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
using System;
using OpenTelemetry;
using OpenTelemetry.Trace;
using System.Diagnostics;
using System.Collections.Generic;
using OpenTelemetry.Resources;
using Grpc.Core;
namespace mydemo
{
    class Program
    {
         private static readonly ActivitySource MyActivitySource = new ActivitySource(
        "MyCompany.MyProduct.MyLibrary");
        static void Main(string[] args) {
            using var tracerProvider = Sdk.CreateTracerProviderBuilder()
           .SetSampler(new AlwaysOnSampler())
           .AddSource("MyCompany.MyProduct.MyLibrary")
           .AddOtlpExporter(opt => {
                opt.Endpoint = new Uri("${endpoint}");
                opt.Headers = "x-sls-otel-project=${project},x-sls-otel-instance-id=${i
nstance},x-sls-otel-ak-id=${access-key-id},x-sls-otel-ak-secret=${access-key-secret}";
           })
            .SetResourceBuilder(OpenTelemetry.Resources.ResourceBuilder.CreateDefault()
           .AddAttributes(new Dictionary<string, object> { { "service.name", "${service
}" },
           {"service.version","${version}"},
           {"service.host","${host}"} }))
           .Build();
            using (var activity = MyActivitySource.StartActivity("SayHello"))
            {
                activity?.SetTag("foo", 1);
                activity?.SetTag("bar", "Hello, World!");
                activity?.SetTag("baz", new int[] { 1, 2, 3 });
            }
            Console.WriteLine("Hello World!");
        }
    }
}
```

```
using System;
using OpenTelemetry;
using OpenTelemetry.Trace;
using System.Diagnostics;
using System.Collections.Generic;
using OpenTelemetry.Resources;
using Grpc.Core;
namespace mydemo
{
    class Program
    {
         private static readonly ActivitySource MyActivitySource = new ActivitySource(
        "MyCompany.MyProduct.MyLibrary");
        static void Main(string[] args) {
            using var tracerProvider = Sdk.CreateTracerProviderBuilder()
           .SetSampler(new AlwaysOnSampler())
           .AddSource("MyCompany.MyProduct.MyLibrary")
           .AddOtlpExporter(opt => {
                opt.Endpoint = new Uri("${endpoint}");
                opt.Headers = "x-sls-otel-project=${project},x-sls-otel-instance-id=${i
nstance},x-sls-otel-ak-id=${access-key-id},x-sls-otel-ak-secret=${access-key-secret}";
           })
            .SetResourceBuilder(OpenTelemetry.Resources.ResourceBuilder.CreateDefault()
           .AddAttributes(new Dictionary<string, object> {
               {"service.name", "${service}" },
                  {"service.version","${version}"},
               {"service.host","${host}"},
               {"service.namespace","${service.namespace}"}
               }))
           .Build();
            using (var activity = MyActivitySource.StartActivity("SayHello"))
            {
                activity?.SetTag("foo", 1);
                activity?.SetTag("bar", "Hello, World!");
                activity?.SetTag("baz", new int[] { 1, 2, 3 });
            }
            Console.WriteLine("Hello World!");
        }
    }
}
```

Variables

Variable	Description	Example
${service}	The name of the service. Specify the value based on your business requirements.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${host}	The hostname.	localhost

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: https://${project}.${region-endpoint}:Port. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. ○ Port: the port number. The value is fixed as 10010.	https://test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	instance-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	LTAI4Fvyv****
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	HfJEw25sYldO****
${service.namespace}	The namespace to which the service belongs.	order

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.6. Import trace data from Rust applications to Log Service by using OpenTelemetry SDK for Rust

This topic describes how to import trace data from Rust applications to Log Service by using OpenTelemetry SDK for Rust.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Rust development environment is set up. The Rust version is 1.46 or later.

Procedure

1. Add dependencies.

```
[package]
name = "test"
version = "0.1.0"
authors = [""]
edition = "2018"
# See more keys and their definitions at The Manifest Format.
[dependencies]
futures = "0.3"
lazy_static = "1.4"
opentelemetry = { version = "0.16.0", features = ["tokio-support", "metrics", "serializ
e"] }
opentelemetry-otlp = { version = "0.9.0", features = ["tonic", "metrics", "tls", "tls-r
oots"] }
serde_json = "1.0"
tokio = { version = "1.0", features = ["full"] }
tonic="0.4.0"
url = "2.2.0"
```

2. Run code.

   Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
use opentelemetry::global::shutdown_tracer_provider;
use opentelemetry::sdk::Resource;
use opentelemetry::trace::TraceError;
use opentelemetry::{
    baggage::BaggageExt,
    trace::{TraceContextExt, Tracer},
    Context, Key, KeyValue,
};
use opentelemetry::{global, sdk::trace as sdktrace};
use opentelemetry_otlp::WithExportConfig;
use std::error::Error;
use std::time::Duration;
use tonic::metadata::MetadataMap;
use tonic::transport::ClientTlsConfig;
use url::Url;
```

```
static ENDPOINT: &str = "https://${endpoint}";
static PROJECT: &str = "${project}";
static INSTANCE_ID: &str = "${instance}";
static AK_ID: &str = "${access-key-id}";
static AK_SECRET: &str = "${access-key-secret}";
static SERVICE_VERSION: &str = "${version}";
static SERVICE_NAME: &str = "${service}";
static SERVICE_NAMESPACE: &str = "${service.namespace}";
static HOST_NAME: &str = "${host}";
static SLS_PROJECT_HEADER: &str = "x-sls-otel-project";
static SLS_INSTANCE_ID_HEADER: &str = "x-sls-otel-instance-id";
static SLS_AK_ID_HEADER: &str = "x-sls-otel-ak-id";
static SLS_AK_SECRET_HEADER: &str = "x-sls-otel-ak-secret";
static SLS_SERVICE_VERSION: &str = "service.version";
static SLS_SERVICE_NAME: &str = "service.name";
static SLS_SERVICE_NAMESPACE: &str = "service.namespace";
static SLS_HOST_NAME: &str = "host.name";
fn init_tracer() -> Result<sdktrace::Tracer, TraceError> {
    let mut metadata_map = MetadataMap::with_capacity(4);
    metadata_map.insert(SLS_PROJECT_HEADER, PROJECT.parse().unwrap());
    metadata_map.insert(SLS_INSTANCE_ID_HEADER, INSTANCE_ID.parse().unwrap());
    metadata_map.insert(SLS_AK_ID_HEADER, AK_ID.parse().unwrap());
    metadata_map.insert(SLS_AK_SECRET_HEADER, AK_SECRET.parse().unwrap());
    let endpoint = ENDPOINT;
    let endpoint = Url::parse(&endpoint).expect("endpoint is not a valid url");
    let resource = vec![
        KeyValue::new(SLS_SERVICE_VERSION, SERVICE_VERSION),
        KeyValue::new(SLS_HOST_NAME, HOST_NAME),
        KeyValue::new(SLS_SERVICE_NAMESPACE, SERVICE_NAMESPACE),
        KeyValue::new(SLS_SERVICE_NAME, SERVICE_NAME),
    ];
    opentelemetry_otlp::new_pipeline()
        .tracing()
        .with_exporter(
            opentelemetry_otlp::new_exporter()
                .tonic()
                .with_endpoint(endpoint.as_str())
                .with_metadata(dbg!(metadata_map))
                .with_tls_config(
                    ClientTlsConfig::new().domain_name(
                        endpoint
                            .host_str()
                            .expect("the specified endpoint should have a valid host"),
                    ),
                ),
        )
        .with_trace_config(sdktrace::config().with_resource(Resource::new(resource)))
        .install_batch(opentelemetry::runtime::Tokio)
}
const FOO_KEY: Key = Key::from_static_str("ex.com/foo");
const BAR_KEY: Key = Key::from_static_str("ex.com/bar");
const LEMONS_KEY: Key = Key::from_static_str("lemons");
const ANOTHER_KEY: Key = Key::from_static_str("ex.com/another");
lazy_static::lazy_static! {
    static ref COMMON_ATTRIBUTES: [KeyValue; 4] = [
```

```
    static ref COMMON_ATTRIBUTES: [KeyValue; 4] = [
        LEMONS_KEY.i64(10),
        KeyValue::new("A", "1"),
        KeyValue::new("B", "2"),
        KeyValue::new("C", "3"),
    ];
}
#[tokio::main]
async fn main() -> Result<(), Box<dyn Error + Send + Sync + 'static>> {
    let _ = init_tracer()?;
    let tracer = global::tracer("ex.com/basic");
    let _baggage =
        Context::current_with_baggage(vec![FOO_KEY.string("foo1"), BAR_KEY.string("bar1
")])
            .attach();
    tracer.in_span("operation", cx {
        let span = cx.span();
        span.add_event(
            "Nice operation!".to_string(),
            vec![Key::new("bogons").i64(100)],
        );
        span.set_attribute(ANOTHER_KEY.string("yes"));
        tracer.in_span("Sub operation...", cx {
            let span = cx.span();
            span.set_attribute(LEMONS_KEY.string("five"));
            span.add_event("Sub span event".to_string(), vec![]);
        });
    });
    tokio::time::sleep(Duration::from_secs(60)).await;
    shutdown_tracer_provider();
    Ok(())
}
```

Variables

Variable	Description	Example
${service}	The name of the service. Specify the value based on your business scenario.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${service.namespace}	The namespace to which the service belongs.	order
${host}	The hostname.	localhost

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}:Port. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints. ○ Port: the port number. The value is fixed as 10010.	test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.7. Import trace data from Ruby applications to Log Service by using OpenTelemetry SDK for Ruby

This topic describes how to import trace data from Ruby applications to Log Service by using OpenTelemetry SDK for Ruby.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A Ruby development environment is set up. The Ruby version is 2.0 or later.
- OpenTelemetry SDK for Ruby is installed.

  If OpenTelemetry SDK for Ruby is not installed, you can run the following commands to install the SDK:

```
gem install opentelemetry-api
gem install opentelemetry-sdk
gem install opentelemetry-exporter-otlp
```

Procedure

1. Configure environment variables.

   Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
export OTEL_RESOURCE_ATTRIBUTES=sls.otel.project=${project},sls.otel.instanceid=${instance},sls.otel.akid=${akid},sls.otel.aksecret=${aksecret},service.name=${service},service.version=${version},host.name=${host}
```

```
export OTEL_RESOURCE_ATTRIBUTES=sls.otel.project=${project},sls.otel.instanceid=${instance},sls.otel.akid=${akid},sls.otel.aksecret=${aksecret},service.namespace=${service.namespace},service.name=${service},service.version=${version},host.name=${host}
```

   Variables

Variable	Description	Example
${service}	The name of the service. Specify the value based on your business requirements.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${service.namespace}	The namespace to which the service belongs.	order
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces

Variable	Description	Example
${akid}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${aksecret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None
${host}	The hostname.	localhost

2. Configure instrumentation.

   Replace the *${endpoint}* variable in the following code with the actual value. For more information about the variables, see Variables. For more information about sample code, see opentelemetry-ruby.

```
require 'opentelemetry/sdk'
require 'opentelemetry-exporter-otlp'
# Configure the sdk with default export and context propagation formats
# see SDK#configure for customizing the setup
OpenTelemetry::SDK.configure do |c|
  c.add_span_processor(
    OpenTelemetry::SDK::Trace::Export::BatchSpanProcessor.new(
      OpenTelemetry::Exporter::OTLP::Exporter.new(
        endpoint: 'https://${endpoint}/opentelemetry/v1/traces'
      )
    )
  )
end
# To start a trace you need to get a Tracer from the TracerProvider
tracer = OpenTelemetry.tracer_provider.tracer('my_app_or_gem', '0.1.0')
tracer.in_span('foo') do |span|
  # set an attribute
  span.set_attribute('tform', 'osx')
  # add an event
  span.add_event('event in bar')
  # create bar as child of foo
  tracer.in_span('bar') do |child_span|
    # inspect the span
    pp child_span
  end
end
sleep 10
```

Variables

Variable	Description	Example
${endpoint}	The endpoint of the Log Service project. Format: ${project}.${region-endpoint}. ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using an internal or public endpoint. An internal endpoint can be accessed over the classic network or a virtual private cloud (VPC). A public endpoint can be accessed over the Internet. For more information, see Endpoints.	https://test-project.cn-hangzhou.log.aliyuncs.com

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.8. Import trace data from PHP applications to Log Service by using Zipkin

This topic describes how to import trace data from PHP applications to Log Service by using Zipkin.

Context

- A trace instance is created. For more information, see Create a trace instance.
- PHP is installed.
- Composer is installed.

Procedure

1. Download the official sample code of Zipkin.
2. Modify the parameters in the *functions.php* file.
   i. Modify the $httpReporterURL parameter.

   Replace the *${endpoint}* variable in the code with the actual value. For more information about the variables, see Variables.

   ```
   $httpReporterURL = 'https://${endpoint}/zipkin/api/v2/spans';
   ```

   Variables

Variable	Description	Example
${endpoint}	The endpoint. The format is ${project}.${region-endpoint}, where: ■ ${project}: the name of the Log Service project. ■ ${region-endpoint}: the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a virtual private cloud (VPC). For more information, see Endpoints.	test-project.cn-hangzhou.log.aliyuncs.com

ii. Add the headers parameter when you create the Zipkin\Reporters\Http file.

Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
$reporter = new Zipkin\Reporters\Http(
    \Zipkin\Reporters\Http\CurlFactory::create(),
    ['endpoint_url' => $httpReporterURL,
           'headers' => ['x-sls-otel-project' => '${project}',
                  'x-sls-otel-instance-id' => '${instance}',
                  'x-sls-otel-ak-id' => '${access-key-id}',
                  'x-sls-otel-ak-secret' => '${access-key-secret}']
    ]
);
```

Variables

Variable	Description	Example
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a Resource Access Management (RAM) user who has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

3. Install dependencies.

```
composer install
```

4. Start the service.

```
composer run-frontend
composer run-backend
```

5. Access the service and then send the trace data to Log Service.

```
curl http://localhost:8081
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.9. Import trace data from C++ applications to Log Service by using Jaeger SDK for C++

This topic describes how to import trace data from C++ applications to Log Service by using Jaeger SDK for C++.

Prerequisites

- A trace instance is created. For more information, see Create a trace instance.
- A development environment in which Jaeger SDK for C++ can be compiled and run is prepared.
  - If you use the CMake Editor, the version must be 3.0 or later.
  - If you use the GCC or g++ compiler, the version must be 4.9.0 or later.

Procedure

1. Download and compile the SDK.
   i. Download Jaeger SDK for C++.
   ii. Decompress the package to the specified path.
   iii. Go to the specified path after the package is decompressed, and run the following commands:

```
mkdir build
cd build
cmake ..
make
```

2. Compile and run the code.
   i. Modify the *App.cpp* file of the *examples* directory.

   Replace the content of the *App.cpp* file with the following content. The following content indicates that Jaeger is initialized by using environment variables. For more information, see jaeger-client-cpp.

```
#include <iostream>
#include <jaegertracing/Tracer.h>
#include <jaegertracing/utils/EnvVariable.h>
namespace {
void setUpTracer()
{
    const auto serviceName = jaegertracing::utils::EnvVariable::getStringVariable("
JAEGER_SERVICE_NAME");
    auto config = jaegertracing::Config();
    config.fromEnv();
    auto tracer = jaegertracing::Tracer::make(
        serviceName, config, jaegertracing::logging::consoleLogger());
    opentracing::Tracer::InitGlobal(
        std::static_pointer_cast<opentracing::Tracer>(tracer));
}
void tracedSubroutine(const std::unique_ptr<opentracing::Span>& parentSpan)
{
    auto span = opentracing::Tracer::Global()->StartSpan(
        "tracedSubroutine", { opentracing::ChildOf(&parentSpan->context()) });
}
void tracedFunction()
{
    auto span = opentracing::Tracer::Global()->StartSpan("tracedFunction");
    tracedSubroutine(span);
}
}  // anonymous namespace
int main(int argc, char* argv[])
{
    setUpTracer();
    tracedFunction();
    // Not stricly necessary to close tracer, but might flush any buffered
    // spans. See more details in opentracing::Tracer::Close() documentation.
    opentracing::Tracer::Global()->Close();
    return 0;
}
```

ii. Go to the *build* directory.

iii. Run the `make` command to build an application.

iv. Run the following code.

You must replace the variables in the following code with the actual values. The following table describes the variables.

```
# If you want to print spans, set the required environment variable in the format o
f export JAEGER_REPORTER_LOG_SPANS=true.
export JAEGER_SAMPLER_TYPE=const
export JAEGER_SAMPLER_PARAM=1
export JAEGER_SERVICE_NAME=${service}
export JAEGER_PROPAGATION=w3c
export JAEGER_ENDPOINT="https://${endpoint}/jaeger/api/traces"
export JAEGER_TAGS=sls.otel.project=${project},sls.otel.instanceid=${instance},sls.
otel.akid=${access-key-id},sls.otel.aksecret=${access-key-secret},service.version=$
{version}
./app
```

Variables

Variable	Description	Example
${service}	The name of the service. Enter a name based on the actual scenario.	payment
${version}	The version of the service. We recommend that you specify the version in the va.b.c format.	v0.1.2
${endpoint}	The endpoint. The format is ${project}.${region-endpoint}, where: ■ ${project}: the name of the Log Service project. ■ ${region-endpoint}: the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a virtual private cloud (VPC). For more information, see Endpoints. ■ Port: the port number, which is set to 10010. **Note** ■ If you set the variable to stdout in the provider.WithTraceExporterEndpoint("stdout"), format, data is printed as standard outputs. ■ If you set the variable to an empty value, trace data is not uploaded to Log Service.	test-project.cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces

Variable	Description	Example
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a Resource Access Management (RAM) user who has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.2.10. Import trace data from Android apps to Log Service

This topic describes how to import trace data from Android apps to Log Service by using Log Service SDK for Android.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Step 1: Import library files

Add the following dependencies to the *build.gradle* file of the related module in Android Studio:

```
implementation 'com.aliyun.openservices:aliyun-log-android-sdk:bricks_1.0.7'
implementation 'com.aliyun.openservices:alysls-android-trace:1.0.0'
```

The following table describes the dependency packages that are required to import trace data from Android apps.

Library file	Description
aliyun-log-android-sdk	Contains the core SDK that is used to import data from Android apps to Log Service.
alysls-android-trace	Contains the plug-in that is used to import trace data.

Step 2: Configure a service to import data

1. Add an Application class to the *$PROJECT/app/src/main/AndroidManifest.xml* file.

   The following example shows how to add the MyApplication class:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          package="com.aliyun.sls.android.demo">
    ...
    <application
        android:icon="@mipmap/ic_launcher"
        ...
        android:name="com.aliyun.sls.android.demo.SLSDemoApplication"
        ...
        android:theme="@style/AppTheme">
            ...
    </application>
</manifest>
```

   Android Studio automatically creates a class named MyApplication based on the MyApplication class that you added, and then adds the new class to the current project.

2. In the MyApplication.onCreate method, add the following initialization code:

```
public class MyApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        SLSAdapter adapter = SLSAdapter.getInstance();
        // Add the Trace plug-in.
        adapter.addPlugin(SLSTracePlugin.getInstance());
        SLSConfig config = new SLSConfig(this);
        config.pluginTraceEndpoint = "endpoint";
        config.pluginTraceLogProject = "project";
        config.pluginTraceLogStore = "store";
        // Enable log debugging based on integration.
        config.debuggable = true;
        adapter.init(config);
    }
}
```

   - SLSConfig

     The following table describes the parameters in the SLSConfig class.

Category	Parameter	Example	Description
Debugging parameter	debuggable	true	Specifies whether to enable log debugging. **Note** When you publish the app, we recommend that you specify config.debuggable = false to disable log debugging.
Configuration parameter	appVersion	None	The version number of the app. We recommend that you use the default settings.
	appName	None	The name of the app. We recommend that you use the default settings.
Configuration parameter	traceEndpoint	cn-hangzhou.log.aliyuncs.com	The endpoint of the project that is specified when you created the trace instance. For more information, see Public Log Service endpoints. **Notice** Only public Log Service endpoints are supported.
	traceLogProject	sls-ayasls-demo	The name of the project that is specified when you created the trace instance. For more information, see Create a trace instance.
	traceLogStore	sls-d60f-traces	The name of the Logstore that is used to store raw trace data in the Trace application. The name of the Logstore is in the {instance}-traces format. The value of the *instance* variable is the ID of the trace instance that you created. For more information, see Create a trace instance.
	accessKeyId	LTAI****eYDw	The AccessKey ID of the RAM user that has the permissions to access the Log Service project. For information about how to obtain an AccessKey pair, see AccessKey pair. For information about how to grant the required permissions to a RAM user, see Configure the permission assistant feature.

Category	Parameter	Example	Description
	accessKeySecret	lrRq****GOVM	The AccessKey secret of the RAM user that has the permissions to access the Log Service project. For information about how to obtain an AccessKey pair, see AccessKey pair. For information about how to grant the required permissions to a RAM user, see Configure the permission assistant feature.
	securityToken	124f****a369	The Security Token Service (STS) token of the RAM user that has the permissions to access the Log Service project. Before you use STS to access the project, you must obtain an STS token. For more information, see AssumeRole.
Custom parameter	channel	Google	The custom parameter that specifies the channel identifier of the app.
	channelName	Google	The custom parameter that specifies the channel name of the app.
	userNick	Tom	The custom parameter that specifies the nickname of the user.
	longLoginNick	Tom	The custom parameter that specifies the user nickname used for the last logon.
	userId	423423	The custom parameter that specifies the ID of the user.
	longLoginUserId	423423	The custom parameter that specifies the user ID used for the last logon.
	loginType	pswd	The custom parameter that specifies the logon type.
Business parameter	slsConfig.addCustom("customKey", "customValue");	slsConfig.addCustom("action_name", "click");slsConfig.addCustom("action_args", "detail_id:243434");	The business parameters in the format of key-value pairs. ■ customKey: the name of the business parameter. ■ customValue: the value of the business parameter.

- SLSAdapter

  The SLSAdapter class is used to manage plug-ins.

Method	Description
getInstance()	Returns an SLSAdapter singleton instance.
addPlugin(plugin)	Adds a plug-in.
init(slsConfig)	Initializes all plug-ins. **Notice** The init() method is called to initialize all plug-ins. Do not call the init() method of a plug-in. Before you call the init() method, you must call the addPlugin() method.
resetSecurityToken(accessKeyId, accessKeySecret, securityToken)	Updates STS tokens. **Notice** Before you call the resetSecurityToken() method, you must call the initWithSLSConfig method.
updateConfig(slsConfig)	Updates the settings of the SLSConfig class. You can call this method to update the traceEndpoint, traceLogProject, and traceLogStore parameters and custom parameters. **Notice** Before you call the updateConfig () method, you must call the initWithSLSConfig method.

3. Use STS to set the accessKeyId, accessKeySecret, and securityToken parameters.

```
public class MyApplication extends Application {
    @Override
    public void onCreate() {
        super.onCreate();
        SLSAdapter adapter = SLSAdapter.getInstance();
        // Add the Trace plug-in.
        adapter.addPlugin(SLSTracePlugin.getInstance());
        SLSConfig config = new SLSConfig(this);
        // Specify the AccessKey pair and STS token when you initialize the code.
        config.accessKeyId = accesskeyid;
        config.accessKeySecret = accesskeysecret;
        config.securityToken = securityToken;
        config.pluginTraceEndpoint = "endpoint";
        config.pluginTraceLogProject = "project";
        config.pluginTraceLogStore = "logstore";
        // Enable log debugging based on integration.
        config.debuggable = true;
        adapter.init(config);
    }
}
```

4. Construct a span.

```
// 1. Obtain the trace instance.
Tracer tracer = SLSTelemetry.getInstance().getTracer("your_tracer_name");
// 2. Construct SpanBuilder.
SpanBuilder spanBuilder = tracer.spanBuilder("your_span_name");
spanBuilder.setAttribute("Attribute key", "Attribute value");
// 3. Construct a span.
Span span = spanBuilder.startSpan();
// Add an attribute based on your business requirements.
span.setAttribute("Attribute key", "Attribute value");
// Add an event based on your business requirements.
span.addEvent("your event");
// 4. Specify the status of the span. Default value: UNSET. Set OK on a completed span.
If the span is not completed, set the status to ERROR.
span.setStatus(StatusCode.OK);
// 5. End the span.
span.end();
```

5. Construct trace data.

```
// 1. Obtain the trace instance.
Tracer tracer = SLSTelemetry.getInstance().getTracer("your_tracer_name");
// 2. Construct the first span.
SpanBuilder spanBuilder = tracer.spanBuilder("your_span_name_1");
spanBuilder.setAttribute("Attribute key1", "Attribute value1");
Span span = spanBuilder.startSpan();
span.setAttribute("Attribute key2", "Attribute value2");
span.addEvent("your event");
span.setStatus(StatusCode.OK);
// 3. Construct the second span.
SpanBuilder spanBuilder = tracer
                          .spanBuilder("your_span_name_2")
                          // Specify the context of the first span.
                          .setParent(Context.current().with(span));
spanBuilder.setAttribute("Attribute key1", "Attribute value1");
Span span2 = spanBuilder.startSpan();
span.setAttribute("Attribute key2", "Attribute value2");
span2.addEvent("your event");
span2.setStatus(StatusCode.OK);
// 4. End the second span.
span2.end();
// 5. End the first span.
span.end();
```

6. Use TextMapPropagator to connect the Android app and the backend.

```
// 1. Obtain the trace instance.
Tracer tracer = SLSTelemetry.getInstance().getTracer("your_tracer_name");
// 2. Construct the first span.
SpanBuilder spanBuilder=tracer.spanBuilder ("/"); // Specify a name for the span.
spanBuilder.setAttribute("Attribute key1", "Attribute value1");
Span span = spanBuilder.startSpan();
span.setAttribute("Attribute key2", "Attribute value2");
span.addEvent("your event");
span.setStatus(StatusCode.OK);
span.end();
// In this example, HttpURLConnection is used.
HttpURLConnection connection = (HttpURLConnection) url.openConnection();
// 3. Construct TextMapSetter.
TextMapSetter<HttpURLConnection> setter = new TextMapSetter<HttpURLConnection>() {
    @Override
    public void set(HttpURLConnection carrier, String key, String value) {
        carrier.setRequestProperty(key, value);
    }
};
// 4. Inject traceparent and tracestate in HTTP request headers.
SLSTelemetry slsTelemetry = SLSTelemetry.getInstance();
TextMapPropagator textMapPropagator = slsTelemetry.getPropagators().getTextMapPropagato
r();
textMapPropagator.inject(context, connection, setter);
```

What's next

- View the details of a trace instance

- Query and analyze trace data

3.4.2.11. Import trace data from iOS apps to Log Service

This topic describes how to import trace data from iOS apps to Log Service by using Log Service SDK for iOS.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Step 1: Import library files

1. Add the following content to the Podfile in Xcode:

```
source 'https://github.com/CocoaPods/Specs.git'
pod 'AliyunLogProducer/Bricks', '~> 2.3.6.1'
pod 'AliyunLogProducer/Trace', '~> 2.3.6.1'
```

   The following table describes the dependency packages that are required when you import trace data from iOS apps.

Library file	Description
AliyunLogProducer/Bricks	Contains the core SDK that is used to import data from iOS applications to Log Service.
AliyunLogProducer/Trace	Contains the plug-in that is used to import trace data.

2. Save and run the `pod install` command.
3. Use a file suffixed by *.xcworkspace* to open Xcode.

Step 2: Configure a service to import data

1. Import the following header file to the *AppDelegate.m* file in Xcode:

```
#import <AliyunLogProducer/AliyunLogProducer.h>
```

2. Initialize the SDK.

```
// Initialize code.
SLSConfig *config = [[SLSConfig alloc] init];
// When you publish the app, we recommend that you specify [config setDebuggable:NO] to
disable log debugging.
[config setDebuggable:YES];
[config setAccessKeyId: [utils accessKeyId]];
[config setAccessKeySecret: [utils accessKeySecret]];
// When you configure the Trace plug-in, you must use the setTrace method.
[config setTraceEndpoint:@"endpoint"];
[config setTraceLogproject:@"project"];
[config setTraceLogstore:@"logstore"];
SLSAdapter *slsAdapter = [SLSAdapter sharedInstance];
[slsAdapter addPlugin:[[SLSTracePlugin alloc] init]];
[slsAdapter initWithSLSConfig:config];
```

The following part describes the SLSConfig and SLSAdapter classes.

- SLSConfig

  The following table describes the parameters in the SLSConfig class.

Category	Parameter	Example	Description
Debugging parameter	debuggable	true	Specifies whether to enable log debugging.
Configuration parameter	appVersion	None	The version number of the app. We recommend that you use the default settings.
	appName	None	The name of the app. We recommend that you use the default settings.
Configuration parameter	traceEndpoint	cn-hangzhou.log.aliyuncs.com	The endpoint of the project that is specified when you created the trace instance. For more information, see Public Log Service endpoints. **Notice** Only public Log Service endpoints are supported.
	traceLogproject	sls-ayasls-demo	The name of the project that is specified when you created the trace instance. For more information, see Create a trace instance.
	traceLogstore	sls-d60f-traces	The name of the Logstore that is used to store raw trace data in the Trace application. The name of the Logstore is in the {instance}-traces format. The value of the *instance* variable is the ID of the trace instance that you created. For more information, see Create a trace instance.
	accessKeyId	LTAI****eYDw	The AccessKey ID of the RAM user that has the permissions to access the Log Service project. For information about how to obtain an AccessKey pair, see AccessKey pair. For information about how to grant the required permissions to a RAM user, see Configure the permission assistant feature.

Category	Parameter	Example	Description
	accessKeySecret	lrRq****GOVM	The AccessKey secret of the RAM user that has the permissions to access the Log Service project. For information about how to obtain an AccessKey pair, see AccessKey pair. For information about how to grant the required permissions to a RAM user, see Configure the permission assistant feature.
	securityToken	124f****a369	The Security Token Service (STS) token of the RAM user that has the permissions to access the Log Service project. Before you use STS to access the project, you must obtain an STS token. For more information, see AssumeRole.
Custom parameter	channel	Apple	The custom parameter that specifies the channel identifier of the app.
	channelName	Apple	The custom parameter that specifies the channel name of the app.
	userNick	Tom	The custom parameter that specifies the nickname of the user.
	longLoginNick	Tom	The custom parameter that specifies the user nickname used for the last logon.
	userId	423423	The custom parameter that specifies the ID of the user.
	longLoginUserId	423423	The custom parameter that specifies the user ID used for the last logon.
	loginType	pswd	The custom parameter that specifies the logon type.
Business parameter	[config addCustomWithKey:@"customKey" andValue:@"testValue"];	[config addCustomWithKey:@"action_name" andValue:@"action_args"];	The business parameters in the format of key-value pairs ■ customKey: the name of the business parameter. ■ testValue: the value of the business parameter.

- SLSAdapter

  The SLSAdapter class is used to manage plug-ins.

Method	Description
addPlugin	Adds a plug-in.
removePlugin	Removes a plug-in.
initWithSLSConfig	Initializes all plug-ins. **Notice** The initWithSLSConfig method is called to initialize all plug-ins. Do not call the initWithSLSConfig method of a plug-in. Before you call the initWithSLSConfig method, you must call the addPlugin method.
resetSecurityToken(accessKeyId, accessKeySecret, securityToken)	Updates STS tokens. **Notice** Before you call the resetSecurityToken method, you must call the initWithSLSConfig method.
updateConfig(slsConfig)	Updates the settings of the SLSConfig class. You can call this method to update the traceEndpoint, traceLogProject, and traceLogStore parameters and custom parameters. **Notice** Before you call the updateConfig method, you must call the initWithSLSConfig method.

3. Use STS to set the AccessKeyId, AccessKeySecret, and SecurityToken parameters.

```
// Initialize code.
SLSConfig *config = [[SLSConfig alloc] init];
// When you publish the app, we recommend that you disable log debugging.
[config setDebuggable:YES];
// Specify the AccessKey pair and STS token when you initialze the code.
[config setAccessKeyId: [utils accessKeyId]];
[config setAccessKeySecret: [utils accessKeySecret]];
[config setSecurityToken: [utils token]];
// When you configure the Trace plug-in, you must use the setTrace method.
[config setTraceEndpoint:@"endpoint"];
[config setTraceLogproject:@"project"];
[config setTraceLogstore:@"logstore"];
SLSAdapter *slsAdapter = [SLSAdapter sharedInstance];
[slsAdapter addPlugin:[[SLSTracePlugin alloc] init]];
[slsAdapter initWithSLSConfig:config];
```

4. Construct a span.

```
// 1. Obtain the trace instance.
TelemetrySDK *sdk = [TelemetrySDK instance];
TelemetryTracer *tracer = [sdk getTracer:@"your_tracer_name"];
// 2. Construct SpanBuilder.
TelemetrySpanBuilder *spanBuilder = [tracer spanBuilderWithSpanName:@"your_span_name"];
[spanBuilder setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
// 3. Construct a span.
TelemetrySpan *span = [spanBuilder startSpan];
// Add an attribute based on your business requirements.
[span setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
// Add an event based on your business requirements.
[span addEventWithName:@"your event"];
// 4. Specify the status of the span. Default value: UNSET. Set OK on a completed span.
If the span is not completed, set the status to ERROR.
[span setStatusWithStatus:TelemetryStatus.OK];
// 5. End the span.
[span end];
```

5. Construct trace data.

```
// 1. Construct the first span.
TelemetrySDK *sdk = [TelemetrySDK instance];
TelemetryTracer *tracer = [sdk getTracer:@"your_tracer_name"];
TelemetrySpanBuilder *spanBuilder = [tracer spanBuilderWithSpanName:@"your_span_name"];
[spanBuilder setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
TelemetrySpan *span = [spanBuilder startSpan];
[span setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
[span addEventWithName:@"your event"];
// 2. Construct the second span.
TelemetrySpanBuilder *spanBuilder2 = [tracer spanBuilderWithSpanName:@"your_span_name"]
;
[spanBuilder2 setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
// 3. Set Parent to the first span.
[spanBuilder2 setParent: span];
TelemetrySpan *span2 = [spanBuilder startSpan];
[span2 setAttributeWithKey:@"Attribute key" stringValue:@"Attribute value"];
[span2 addEventWithName:@"your event"];
// 4. End the second span.
[span2 end];
// 5. End the first span.
[span end];
```

6. Use TextMapPropagator to connect the iOS app and the backend.
   i. Construct a TelemetrySetter class.

```
@interface Setter : NSObject  <TelemetrySetter>
@end
@implementation Setter
- (void)set:(NSMutableDictionary * _Nonnull)dict :(NSString * _Nonnull)key :(NSStri
ng * _Nonnull)value {
    [dict setObject:value forKey:key];
}
@end
```

ii. Use TextMapPropagator to generate a dictionary whose key is traceparent.

```
TelemetrySDK *sdk = [TelemetrySDK instance];
TelemetrySpan *span = [[[sdk getTracer:@"demo"] spanBuilderWithSpanName:@"/"] start
Span];
NSMutableDictionary *dict = [NSMutableDictionary dictionary];
// After you call the following method, the dictionary contains a key-value pair wh
ose key is traceparent.
// Add the key-value pair to URLRequest.allHTTPHeaderFields.
[[sdk activeTextMapPropagator] injectWithContext:span.context carrier:dict setter:[
[Setter alloc] init]];
[span end];
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.3. Integrate existing import methods

3.4.3.1. Import trace data from OpenCensus to Log Service

You can send trace data from OpenCensus to the OpenTelemetry Collector by using OpenCensus SDK, and then forward the data to Log Service by using the OpenTelemetry Collector. This topic describes how to forward trace data to Log Service by using the OpenTelemetry Collector.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Procedure

1. Install the OpenTelemetry Collector.
   i. Download the OpenTelemetry Collector.
   ii. Configure the OpenTelemetry Collector.
      a. Create a file named *config.yaml*.
      b. Add the following code to the *config.yaml* file.

         Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
receivers:
  opencensus:
    endpoint: 0.0.0.0:6850
exporters:
  logging/detail:
    loglevel: debug
  alibabacloud_logservice/trace:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance}-traces"
    access_key_id: "${access-key-id}"
    access_key_secret: "${access-key-secret}"
service:
  pipelines:
    traces:
      receivers: [opencensus]    # Set the receivers parameter to opencensus.
      exporters: [alibabacloud_logservice/trace]    # Set the exporters paramet
er to alibabacloud_logservice/trace.
      # for debug
      #exporters: [logging/detail,alibabacloud_logservice/trace]
```

Variables

Variable	Description	Example
${endpoint}	The endpoint of Log Service. The format is ${region-endpoint}. ${region-endpoint} is the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a virtual private cloud (VPC). For more information, see Endpoints.	cn-hangzhou.log.aliyuncs.com
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a Resource Access Management (RAM) user who has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None

Variable	Description	Example
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

iii. Start the OpenTelemetry Collector.

```
./otelcontribcol_linux_amd64 --config="./config.yaml"
```

2. Configure OpenCensus.

   Change the endpoint of OpenCensus to the endpoint on which the OpenTelemetry Collector listens. For example, if the endpoint of the OpenTelemetry Collector is *${collector-host}*, you must set the endpoint of OpenCensus to *${collector-host}*:6850.

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.3.2. Import trace data from Zipkin to Log Service

You can import trace data from Zipkin to Log Service. You can also use the OpenTelemetry Collector to forward trace data to Log Service.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Import trace data from Zipkin

If you want to use the Zipkin protocol to import trace data to Log Service, you must configure the endpoint and authentication settings in the Zipkin SDK. The following list describes the settings:

> **Notice** To ensure data security during transmission, you must import data over HTTPS.

- Endpoint settings
  - HTTP 2.0: An HTTPS endpoint is in the ${endpoint}/zipkin/api/v2/spans format. Example: https://test-project.cn-hangzhou-intranet.log.aliyuncs.com/zipkin/api/v2/spans. We recommend that you use this type of endpoint.
  - HTTP 1.0: An HTTPS endpoint is in the ${endpoint}/zipkin/api/v1/spans format. Example: https://test-project.cn-hangzhou.log.aliyuncs.com/zipkin/api/v1/spans.

  You must replace the *${endpoint}* variable with the actual value. The following table describes the variable.

  Variable description

Variable	Description	Example
${endpoint}	The endpoint. The format is ${project}.${region-endpoint}, where: ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the Log Service endpoint for the region where the project resides. You can access Log Service by using a public or internal endpoint. A public endpoint is accessible over the Internet. An internal endpoint is accessible over the classic network or a virtual private cloud (VPC). For more information, see Endpoints.	test-project.cn-hangzhou.log.aliyuncs.com

- Authentication settings

  You can configure the authentication settings in HTTPS header fields. The following table describes the fields.

HTTPS header field	Description	Example
x-sls-otel-project	The name of the Log Service project.	test-project
x-sls-otel-instance-id	The ID of the trace instance.	test-traces
x-sls-otel-ak-id	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
x-sls-otel-ak-secret	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None

Use the OpenTelemetry Collector to forward trace data

You can use the Zipkin SDK to import trace data from Zipkin to the OpenTelemetry Collector, and then use the OpenTelemetry Collector to forward the data to Log Service. This method supports data transmission over HTTP or HTTPS.

1. Install the OpenTelemetry Collector.
   i. Download the OpenTelemetry Collector.
   ii. Configure the OpenTelemetry Collector.

a. Create a file named *config.yaml*.

b. Add the following code to the *config.yaml* file.

Replace the variables in the following code with the actual values. For more information about the variables, see Variable description.

```
receivers:
  zipkin:
    endpoint: 0.0.0.0:9411
exporters:
  logging/detail:
    loglevel: debug
  alibabacloud_logservice/sls-trace:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance}-traces"
    access_key_id: "${access-key-id}"
    access_key_secret: "${access-key-secret}"
service:
  pipelines:
    traces:
      receivers: [zipkin]              # Set the receivers parameter to zipkin.

      exporters: [alibabacloud_logservice/sls-trace]   # Set the exporters para
meter to alibabacloud_logservice/sls-trace.
      # for debug
      #exporters: [logging/detail,alibabacloud_logservice/sls-trace]
```

Variable description

Variable	Description	Example
${endpoint}	The endpoint of Log Service. The format is ${region-endpoint}. ${region-endpoint} is actually the endpoint for the region where the Log Service project resides. You can access Log Service by using a public or internal endpoint. A public endpoint is accessible over the Internet. An internal endpoint is accessible over the classic network or a VPC. For more information, see Endpoints.	cn-hangzhou.log.aliyuncs.com
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces

Variable	Description	Example
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. For more information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.	None

iii. Start the OpenTelemetry Collector.

```
./otelcontribcol_linux_amd64 --config="./config.yaml"
```

2. Configure Zipkin.

   Change the output endpoint of Zipkin to an endpoint on which the OpenTelemetry Collector can listen. For example, if the endpoint of the OpenTelemetry Collector is *${collector-host}*, change the output endpoint of Zipkin to *${collector-host}*:9411.

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.3.3. Import trace data from SkyWalking

This topic describes how to forward trace data from SkyWalking to Log Service by using Logtail.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Limits

Only the GRPC protocol of SkyWalking V3 is supported. The release version of SkyWalking is 8.0.0 and later.

Step 1: Configure data import

1.
2. In the **Import Data** section, select **SkyWalking**.
3. Select the Project and the ${instance}-Traces Logstore where your trace instance resides.

4. Create a machine group.
   - If a machine group is available, click **Using Existing Machine Groups**.
   - If no machine groups are available, perform the following steps to create a machine group. In this example, an Elastic Compute Service (ECS) instance is used.
     a. On the **ECS Instances** tab, select Manually Select Instances. Then, select the ECS instance that you want to use and click **Execute Now**.

        For more information, see Install Logtail on ECS instances.

        > **Note** If you want to collect logs from self-managed clusters or servers from third-party cloud service providers, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

     b. After Logtail is installed, click **Complete Installation**.
     c. In the **Create Machine Group** step, configure **Name** and click **Next**.

        Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
5.
6. In the **Specify Data Source** step, add the following parameters and click **Next**.

   ${instance} is the ID of your trace instance. We recommend that you replace this parameter based on your actual scenario.

   > **Note** If your Logtail local port 11800 is occupied, you can replace it with another available port.

```
{
    "inputs" : [
        {
            "detail" : {
                "Address" : "0.0.0.0:11800"
            },
            "type" : "service_skywalking_agent_v3"
        }
    ],
    "aggregators" : [
        {
            "detail" : {
                "MetricsLogstore" : "${instance}-metrics",
                "TraceLogstore" : "${instance}-traces"
            },
            "type" : "aggregator_skywalking"
        }
    ],
    "global" : {
        "AlwaysOnline" : true,
        "DelayStopSec" : 300
    }
}
```

Step 2: Configure the SkyWalking client

Configure the SkyWalking client to send data to the address of the Logtail listener. See the following details:

- If you are using Java Agent, replace the value of the collector.backend_service parameters. For more information, see Agent configuration.
- If you are using .net core Agent, use the `dotnet skyapm config *${service}${endpoint}*` command to generate the configuration file. In this case, you must replace the *${service}* variable with the actual service name. You must replace the *${endpoint}* variable with the machine group IP address and the port number that is configured in Step . For more information, see SkyAPM-donet.
- If you are using another agent or SDK to send data, replace the backend address with the machine group IP address and the port number that is configured in Step .

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.3.4. Import trace data from OpenTelemetry to Log Service

You can import trace data from OpenTelemetry to Log Service, or forward the trace data to Log Service by using the OpenTelemetry Collector.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Import trace data from OpenTelemetry

When you use the OpenTelemetry protocol to import trace data to Log Service, you must configure the endpoint and authentication settings in OpenTelemetry. The following examples describe the required settings:

- Endpoint settings
  - An HTTPS endpoint is in the ${endpoint}/opentelemetry/v1/traces format, for example, https://test-project.cn-hangzhou-intranet.log.aliyuncs.com/opentelemetry/v1/traces.
  - A gRPC endpoint is in the ${endpoint}:10010 format, for example, test-project.cn-hangzhou-intranet.log.aliyuncs.com:10010.

    > **Notice** To ensure transmission security, you must enable Transport Layer Security (TLS) when you use the gRPC protocol.

  You must replace the *${endpoint}* variable with the actual value. The following table describes the variable.

  Variables

Variable	Description	Example
${endpoint}	The endpoint. The format is ${project}.${region-endpoint}, where: ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a virtual private cloud (VPC). For more information, see Endpoints.	test-project.cn-hangzhou.log.aliyuncs.com

- Authentication settings

  You can configure the authentication settings in the header of the gRPC or HTTPS protocol, or in the Resource field of the OpenTelemetry protocol. The following table describes the required fields.

OpenTelemetry Resource	gRPC/HTTPS Header Key	Description	Example
sls.otel.project	x-sls-otel-project	The name of the Log Service project.	test-project
sls.otel.instanceid	x-sls-otel-instance-id	The ID of the trace instance.	test-otel

OpenTelemetry Resource	gRPC/HTTPS Header Key	Description	Example
sls.otel.akid	x-sls-otel-ak-id	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a Resource Access Management (RAM) user who has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
sls.otel.aksecret	x-sls-otel-ak-secret	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

Forward trace data by using the OpenTelemetry Collector

1. Download the OpenTelemetry Collector.
2. Configure the OpenTelemetry Collector.
   i. Create a file named *config.yaml*.
   ii. Add the following code to the *config.yaml* file.

   Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
receivers:
  otlp:
    protocols:
      grpc:
        endpoint: "0.0.0.0:55680"
      http:
        endpoint: "0.0.0.0:55681"
exporters:
  logging/detail:
    loglevel: debug
  alibabacloud_logservice/traces:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance-id}-traces"
    access_key_id: "${access-key-id}"
    access_key_secret: "${access-key-secret}"
  alibabacloud_logservice/metrics:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance-id}-metrics"
    access_key_id: "${access-key-id}"
    access_key_secret: "${access-key-secret}"
   alibabacloud_logservice/logs:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance-id}-logs"
    access_key_id: "${access-key-id}"
    access_key_secret: "${access-key-secret}"
service:
  pipelines:
    traces:
      receivers: [otlp]           # Set the receivers parameter to otlp.
      exporters: [alibabacloud_logservice/traces]   # Set the exporters parameter t
o alibabacloud_logservice/traces.
      # for debug
      #exporters: [logging/detail,alibabacloud_logservice/traces]
    metrics:
      receivers: [otlp]
      exporters: [alibabacloud_logservice/metrics]
    logs:
      receivers: [otlp]
      exporters: [alibabacloud_logservice/logs]
```

Variables

Variable	Description	Example
${endpoint}	The endpoint of Log Service. The format is ${region-endpoint}. ${region-endpoint} is the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a VPC. For more information, see Endpoints.	cn-hangzhou.log.aliyuncs.com

Variable	Description	Example
${project}	The name of the Log Service project.	test-project
${instance-id}	The ID of the trace instance.	test-traces
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

3. Start the OpenTelemetry Collector.

```
./otelcontribcol_linux_amd64 --config="./config.yaml"
```

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.3.5. Import trace data from Jaeger to Log Service

You can import trace data from Jaeger to Log Service, or forward the trace data to Log Service by using the OpenTelemetry Collector.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Import trace data from Jaeger

When you use the Jaeger protocol to import trace data to Log Service, you must configure the endpoint and authentication settings in the Jaeger tracing system. The following examples describe the required settings:

- Endpoint settings
  - An HTTPS endpoint is in the ${endpoint}/jaeger/api/traces format, for example, https://test-project.cn-hangzhou-intranet.log.aliyuncs.com/jaeger/api/traces.

- A gRPC endpoint is in the ${endpoint}:10010 format, for example, test-project.cn-hangzhou-intranet.log.aliyuncs.com:10010.

  > **Notice** To ensure transmission security, you must enable Transport Layer Security (TLS) when you use the gRPC protocol.

  You must replace the *${endpoint}* variable with the actual value. The following table describes the variable.

  Variables

Variable	Description	Example
${endpoint}	The endpoint. The format is ${project}.${region-endpoint}, where: ○ ${project}: the name of the Log Service project. ○ ${region-endpoint}: the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a virtual private cloud (VPC). For more information, see Endpoints.	test-project.cn-hangzhou.log.aliyuncs.com

- Authentication settings

  You can configure the authentication settings in the header of the gRPC or HTTPS protocol, or in the Tag field of the Jaeger protocol. The following table describes the required fields.

Jaeger Tag	gRPC/HTTPS Header Key	Description	Example
sls.otel.project	x-sls-otel-project	The name of the Log Service project.	test-project
sls.otel.instanceid	x-sls-otel-instance-id	The ID of the trace instance.	test-traces

Jaeger Tag	gRPC/HTTPS Header Key	Description	Example
sls.otel.akid	x-sls-otel-ak-id	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a Resource Access Management (RAM) user who has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
sls.otel.aksecret	x-sls-otel-ak-secret	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

Forward trace data by using the OpenTelemetry Collector

1. Install the OpenTelemetry Collector.
   i. Download the OpenTelemetry Collector.
   ii. Configure the OpenTelemetry Collector.
      a. Create a file named *config.yaml*.
      b. Add the following code to the *config.yaml* file.

         Replace the variables in the following code with the actual values. For more information about the variables, see Variables.

```
receivers:
  jaeger:
    protocols:
      grpc:
        endpoint: 0.0.0.0:6831
      thrift_binary:
        endpoint: 0.0.0.0:6832
      thrift_compact:
        endpoint: 0.0.0.0:6833
      thrift_http:
        endpoint: 0.0.0.0:6834
exporters:
  logging/detail:
    loglevel: debug
  alibabacloud_logservice/sls-trace:
    endpoint: "${endpoint}"
    project: "${project}"
    logstore: "${instance}-traces"
    access_key_id: "${service}"
    access_key_secret: "${access-key-secret}"
service:
  pipelines:
    traces:
      receivers: [jaeger]        # Set the receivers parameter to jaeger.
      exporters: [alibabacloud_logservice/sls-trace]       # Set the exporters
parameter to alibabacloud_logservice/sls-trace.
      # for debug
      #exporters: [logging/detail,alibabacloud_logservice/sls-trace]
```

Variables

Variable	Description	Example
${endpoint}	The endpoint of Log Service. The format is ${region-endpoint}. ${region-endpoint} is the endpoint of the project. You can access Log Service by using an endpoint of the Internet, the classic network, or a VPC. For more information, see Endpoints.	cn-hangzhou.log.aliyuncs.com:10010
${project}	The name of the Log Service project.	test-project
${instance}	The name of the trace instance.	test-traces

Variable	Description	Example
${access-key-id}	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project. For information about how to grant the write permissions on a specific project to a RAM user, see Use custom policies to grant permissions to a RAM user. For information about how to obtain an AccessKey pair, see AccessKey pair.	None
${access-key-secret}	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user who has only the write permissions on the Log Service project.	None

iii. Start the OpenTelemetry Collector.

```
./otelcontribcol_linux_amd64 --config="./config.yaml"
```

2. Configure Jaeger.

   Change the endpoint of Jaeger to the endpoint on which the OpenTelemetry Collector listens. For example, if the endpoint of the OpenTelemetry Collector is *${collector-host}*, you must set the endpoint of Jaeger to *${collector-host}*:6833.

   > **Note** If an error is returned by the OpenTelemetry Collector because data fails to be parsed, you can switch the four receiving modes of the Jaeger receiver to fix the error.

What's next

- View the details of a trace instance
- Query and analyze trace data

3.4.4. View the import results of trace data

After you import trace data to Log Service, you can query the related logs to check whether trace data is imported to Log Service.

Procedure

1.
2. Open the **Custom Query** page.
   i. In the **Log Application** section, click **Trade Service**.
   ii. In the trace instance list, click the specific instance.
   iii. In the left-side navigation pane, click **Custom Query**.

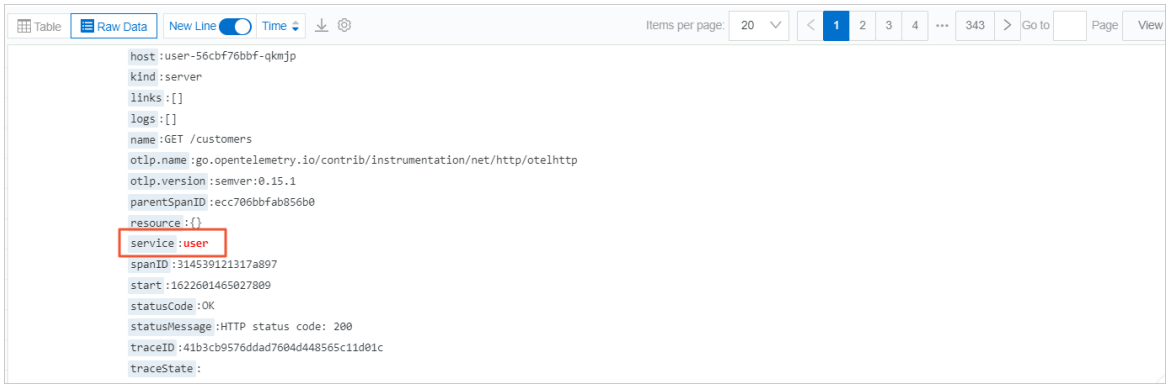
3. Enter a query statement.

   For example, to check whether trace data that is related to the user service is imported, execute the following query statement:

   ```
   service:user
   ```

4. View the query result.

   If the query result contains logs of the user service, you have imported the trace data that is related to the user service.



3.5. View the details of a trace instance

After you import trace data to Log Service, you can view the service details, trace details, and trace dashboards on the details page of a trace instance.

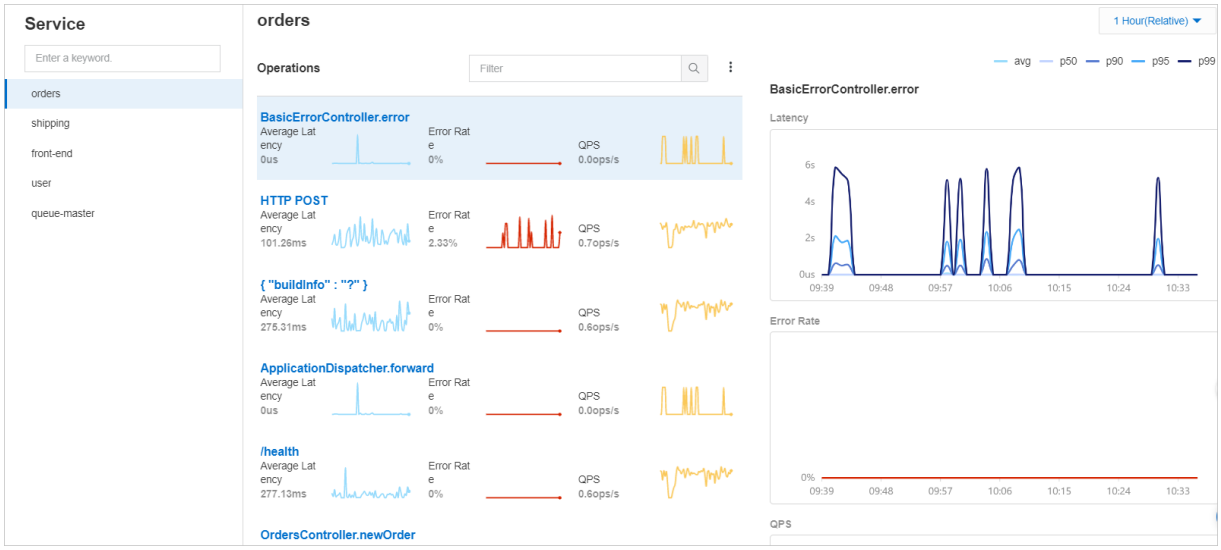
Prerequisites

Trace data is imported. For more information, see Import trace data.

View service details

After you import the trace data of an application to Log Service, you can view all services and service calls of the application in the service list. You can view information such as QPS, error rate, average latency, and P50 latency. You can also filter services and service operations by time range.

For example, if you want to monitor the status of a marketplace system, you can import the trace data of the system to Log Service. All services and service calls of the system are displayed in the service list. The services include user, order, frontend, queue-master, shipping, and job-server.



View trace details

On the **Trace Details** page, you can view trace details, including the trace trail map and span data. For more information, see View the details of a trace.

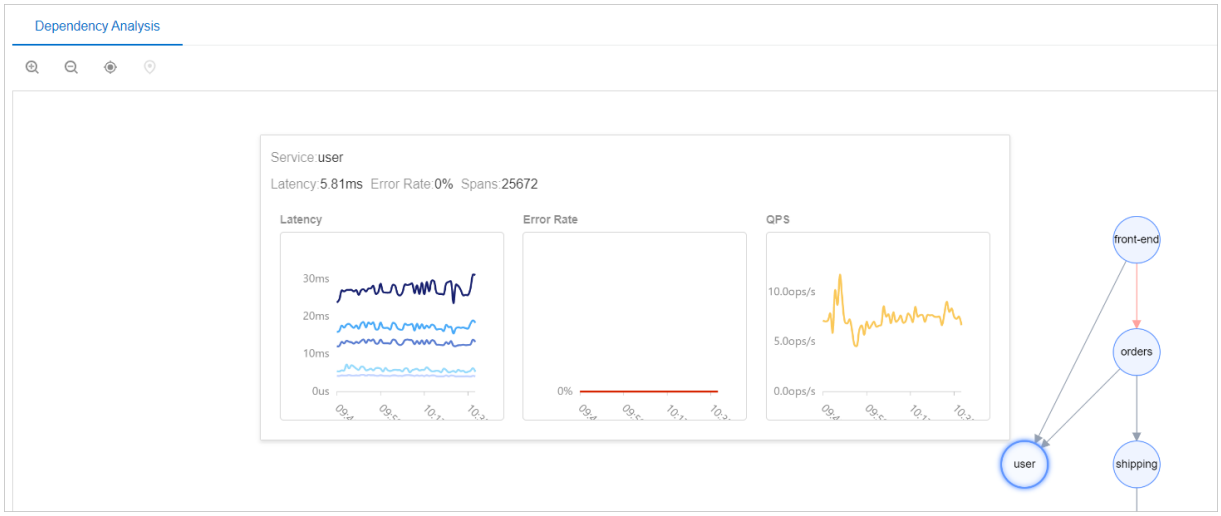
Query and analyze trace data

On the **Trace Analysis** page, you can set query conditions to filter trace data. You can also obtain statistics by service group. For more information, see Query and analyze trace data.

View the topological relationship of services

The topology of services shows the dependencies among these services. You can view the topological relationships among these services on the **Trace Analysis > Dependency Analysis** tab or on the **Topology Query** page.

- After you select a service, click the icon. The system shows only the services that depend on the selected service.
- Click a service. The system shows the average latency, error rate, and number of spans of the service.



Query trace logs

Log Service records the logs that are generated when you import trace data. You can query and analyze the logs on the **Search & Analysis** page. For more information, see Query and analyze logs.

View the trace dashboards

By default, Log Service provides two dashboards. The following table describes the dashboards.

Dashboard	Description
Import Overview	Displays the analysis results of trace data. The results include the number of spans, average latency, services with top 10 latency, services with top 10 requests, and top latency methods.
Statistics	Displays the analysis results of trace data. The results include the number of spans, average latency, services with top 10 latency, and services with top 10 requests.

View storage settings

After you create a trace instance and import trace data to Log Service, Log Service generates a Logstore and a Metricstore to store the trace data, for example, {instance}-traces Logstore and {instance}-metrics Metricstore. For more information, see Assets.

In the **Storage Setting** section, you can view the property of the Logstore or Metricstore.

3.6. Query and analyze trace data

After you import trace data, you can query the trace data and set statistics by group.

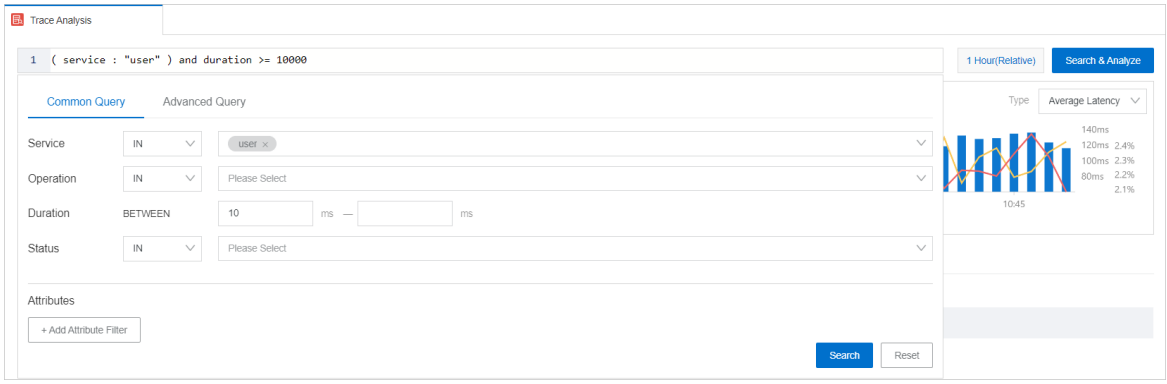
Procedure

1.
2. Open the Trace Analysis page.
   i. In the **Log Application** section, click **Trace Service**.
   ii. In the trace instance list, click the specified instance.
   iii. In the left-side navigation bar, click Trace Analysis.
3. Set query conditions, select a time range, and then click **Search & Analyze**.

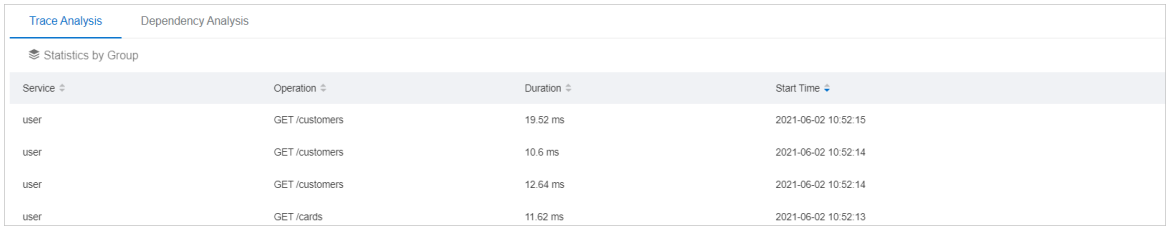
   Log Service provides preset query conditions for multiple fields, such as Service, Operation, Duration, Status, Attribute, and Resource. You only need to select these query conditions based on your business requirements. For more information, see Trace data formats.

> **Note**
>
> - The value of the **Duration** parameter in the query condition is measured in milliseconds. However, the unit of the condition expression is microseconds. For example, if you set the **Duration** parameter to 10 ms, this value is displayed as **duration >= 10000** in the filter condition.
> - By default, the value of the **Duration** parameter is a left-closed and right-open interval.
> - The **Attribute** and **Resource** fields are of the JSON object type. Log Service allows you to filter the fields by key and value in this field.

For example, you want to query the trace data of the last hour for the user service that has a latency of more than 10 ms. You can set the filter conditions shown in the following figure.

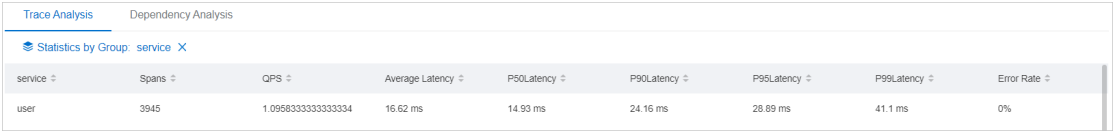


4. On the **Trace Analysis** tab, view the query results.



5. Set statistics by group
   i. On the **Trace Analysis** tab, click **Statistics by Group**.
   ii. Select a condition for the statistics. In this example, select **service**.
   iii. View the statistical results.

   Log Service lists the information of each service by **service**, for example, the number of spans, queries per second (QPS), and average latency.

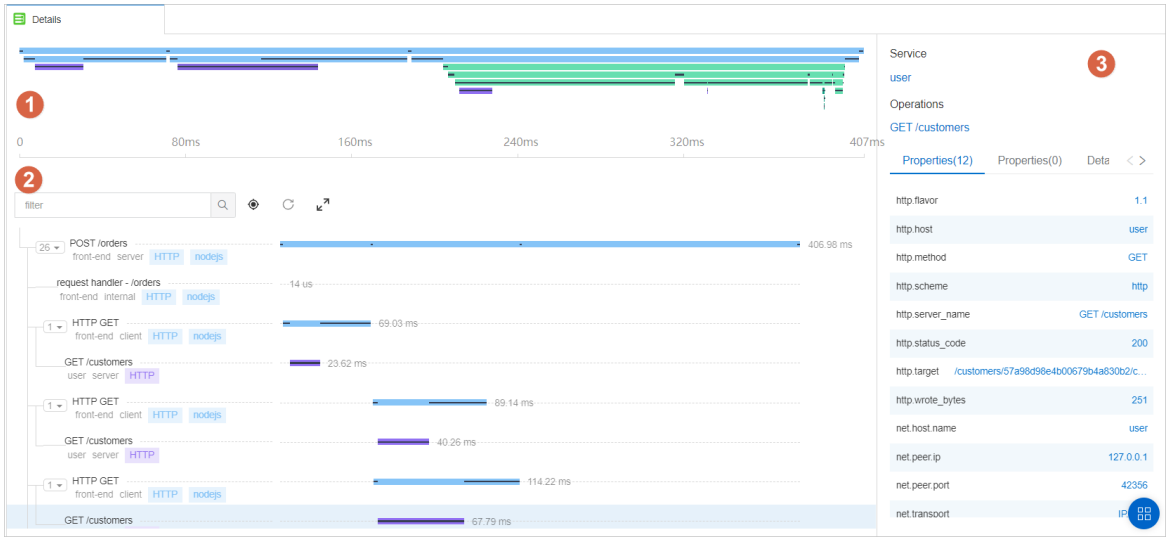


3.7. View the details of a trace

After you import trace data to Log Service, you can view the trace data, including the trace trail map and span data.

Procedure

1.
2. Open the Trace Analysis page.
   i. In the **Log Application** section, click **Trace Service**.
   ii. In the trace instance list, click the specified instance.
   iii. In the left-side navigation bar, click Trace Analysis.
3. On the **Trace Analysis** tab, click the specific trace data.
4. View the details of the trace.



Ordinal number	Description
1	The trace map shows the distribution of tracking links and spans. The following list describes the details: ○ Different colors represent different services. In this example, the blue color represents the front-end service. ○ The length of the black line in the trace map represents the time that is consumed by a span. ○ The timeline represents the time range of the trace.
2	Each row in this section represents a span and shows the hierarchical relationship between the parent span and the child span. Each parent span is preceded by a number, which indicates the number of child spans owned by the parent span. In this section, you can perform the following operations: ○ Click the 17 icon to collapse or expand a span. ○ Select a span and click the icon. The system displays only the data of the span. ○ Click the icon to defocus a span.

Ordinal number	Description
3	When you click a span, the following information is displayed: ○ Service: the name of the service to which the span belongs. ○ Call: the name of the API operation. ○ Property: the metadata of the span. ○ Resource: the resources that must be called to generate the span. ○ Details: the details of the span. For more information about field descriptions, see Trace data formats. ○ Log: the log content that is related to the span.

3.8. Best practices

3.8.1. Import trace data from Log Service to Grafana

Grafana provides a comprehensive user interface (UI). This topic describes how to import trace data from Log Service to Grafana for visualized analysis.

Prerequisites

- Grafana 8.0.0 or a later version is installed. For more information, see Install Grafana.

  > **Note** In this topic, Grafana 8.0.6 that runs on a Linux operating system is used as an example.

- Trace data is collected. For more information, see Overview.
- The project package that contains the Log Service plug-in is downloaded.

Step 1: Install the Log Service plug-in

The following procedure describes how to install the Log Service plug-in for Grafana:

1. Run the following commands to decompress the project package to the plug-in directory of Grafana.
   - If Grafana is installed by using a YUM repository or an RPM package, run the following command:

     ```
     unzip aliyun-log-grafana-datasource-plugin-master.zip -d /var/lib/grafana/plugins
     ```

   - If Grafana is installed by using a *.tar.gz* file, run the following command:

     {PATH_TO} specifies the installation directory of Grafana.

     ```
     unzip aliyun-log-grafana-datasource-plugin-master.zip -d {PATH_TO}/grafana-8.0.6/data
     /plugins
     ```

2. Modify the configuration file of Grafana.

i. Open the configuration file.

- If Grafana is installed by using a YUM repository or an RPM package, open the */etc/grafana/grafana.ini* file.
- If Grafana is installed by using a *.tar.gz* file, open the *{PATH_TO}/grafana-8.0.6/conf/defaults.ini* file.

ii. Find [plugins] in the configuration file to configure the allow_loading_unsigned_plugins parameter.

```
allow_loading_unsigned_plugins = aliyun-log-service-datasource
```

3. Restart the Grafana service.

i. Run the **kill** command to terminate the Grafana process.

ii. Run the following commands to start the Grafana service:

- If Grafana is installed by using a YUM repository or an RPM package, run the following command:

```
systemctl restart grafana-server
```

- If Grafana is installed by using a *.tar.gz* file, run the following command:

```
./bin/grafana-server web
```

Step 2: Add a data source for Grafana

The following procedure describes how to add the Log Service plug-in as a data source for Grafana:

1. Log on to Grafana.
2. In the left-side navigation pane, choose  > **Data Sources**.
3. On the **Data Sources** tab, click **Add data source**.
4. On the **Add data source** page, click **Select** in the **LogService** card.
5. Configure the data source.

   The following table describes the parameters.

Parameter	Description
Name	The name of the data source.
Default	The Default switch. In this example, turn on the switch.
Endpoint	The endpoint of the Log Service project. Example: `http://cn-qingdao.log.aliyuncs.com`. Enter an endpoint based on your business requirements. For more information, see Endpoints.
Project	The name of the project.
Logstore	The name of the Logstore.

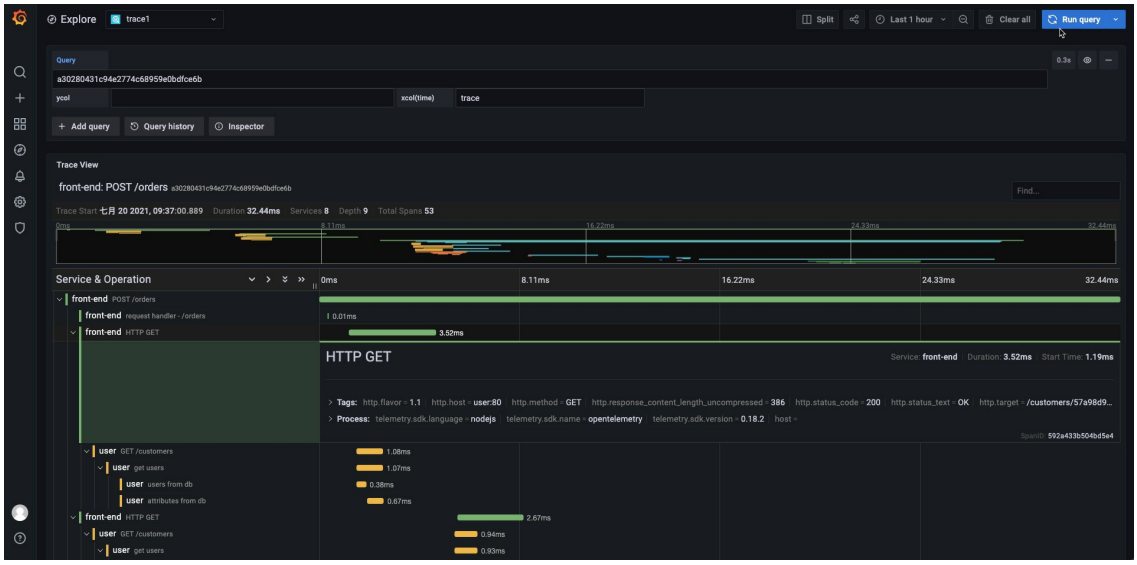
Parameter	Description
AccessKeyId	The AccessKey ID provided by Alibaba Cloud. The AccessKey ID is used to identify the user. To ensure the security of your account, we recommend that you use the AccessKey pair of a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.
AccessKeySecret	The AccessKey secret provided by Alibaba Cloud. The AccessKey secret is used to authenticate the key of the user. To ensure the security of your account, we recommend that you use the AccessKey pair of a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.

6. Click **Save & Test**.

Step 3: View imported trace data

The following procedure describes how to view the trace data imported from Log Service:

1. In the left-side navigation pane, choose > **Explore**.
2. In the upper-left corner of the **Explore** page, select the data source.
3. Enter *trace* in the **xcol(time)** field. Then, click **Run query** in the upper-right corner.



3.8.2. Import trace data from Apache SkyWalking to Log Service

This topic describes how to import trace data from Apache SkyWalking to Log Service. This way, you can query and analyze the trace data by using Log Service.

Prerequisites

- Apache SkyWalking
  - A SkyWalking agent of version 8.0.0 or later is installed in the application on which data is

collected. For more information, see Setup.

- Log Service
  - A Logstore is created. For more information, see Create a Logstore.
  - A machine group is created, and the group uses a custom identifier. For more information, see Create a custom ID-based machine group.

    > **Notice** Make sure that the custom identifier is unique in the region of the Log Service project to which the Logstore belongs.

  - A trace instance is created. For more information, see Create a trace instance.

Context

We recommend that you import trace data from Apache SkyWalking to Log Service. This brings the following benefits:

- Elasticity: Log Service can handle traffic spikes in an efficient manner.
- Performance: Log Service provides higher query performance than open source Elasticsearch. Log Service allows you to write petabytes of data per day, and returns results to queries for billions or tens of billions of data rows within seconds.
- Stability: Log Service uses three replicas for storage, which provides 99.9% availability and 99.999999999% (eleven 9's) reliability.
- O&M: Log Service provides an out-of-the-box feature that allows you to import Apache SkyWalking trace data. O&M is not required for Log Service. You do not need to perform O&M on servers or applications.

Procedure

The following procedure describes how to import trace data from Apache SkyWalking to Log Service:

1.
2. In the **Import Data** section, click **SkyWalking**.
3. In the **Specify Logstore** step, select the project and Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, click **Use Existing Machine Groups**.
5. In the **Machine Group Settings** step, select the machine group that you want to use in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.
6. In the **Specify Data Source** step, configure the Logtail plug-ins and click **Next**.

   A configuration template is provided in the **Plug-in Config** field. You must replace *${instance}* with the ID of your trace instance.

   > **Note** If the local port 11800 of your Logtail is occupied, you can use another available port. In this case, you must change the port that is used by Apache SkyWalking to send data.

```
{
      "inputs" : [
        {
            "detail" : {
                "Address" : "0.0.0.0:11800"
            },
            "type" : "service_skywalking_agent_v3"
        }
    ],
    "aggregators" : [
        {
            "detail" : {
                "MetricsLogstore" : "${instance}-metrics",
                "TraceLogstore" : "${instance}-traces"
            },
            "type" : "aggregator_skywalking"
        }
    ],
    "global" : {
        "AlwaysOnline" : true,
        "DelayStopSec" : 300
    }
}
```

Click **Next** to complete the Logtail configuration. Then, Log Service starts to collect logs.

> **Note**
>
> - A Logtail configuration requires no more than 3 minutes to take effect.
> - If an error occurs when a Logtail configuration is used to collect logs, follow the instructions in How do I view Logtail collection errors? to troubleshoot the error.

What's next

After you import trace data from Apache SkyWalking to Log Service, you can perform the following operations:

- View the details of a trace instance
- Query and analyze trace data
- View the details of a trace

3.8.3. Collect trace data from Apache SkyWalking to Log Service based on an ACK cluster

This topic describes how to collect trace data from Apache SkyWalking by using a Logtail that is deployed in an Alibaba Cloud Container Service for Kubernetes (ACK) cluster to Log Service. After the trace data is collected to Log Service, you can store, analyze, visualize the data in Log Service. You can also configure alerts and perform AIOps based on the data.

Prerequisites

- A Logstore is created. For more information, see Create a Logstore.
- A machine group that has a custom identifier is created. For more information, see Create a custom ID-based machine group.

  > **Notice** Make sure that the custom identifier is unique in the region of the Log Service project to which the Logstore belongs.

- A trace instance is created. For more information, see Create a trace instance.

Step 1: Deploy a data collection image

1. Deploy the Logtail image to an ACK cluster by using a configuration file.

   The following sample code provides a configuration template. For more information, see Create an application by using a private image repository.

```
      containers:
      - name: logtail
        # more info: https://cr.console.aliyun.com/repository/cn-hangzhou/log-service/l
ogtail/detail
        image: registry.cn-hangzhou.aliyuncs.com/log-service/logtail:v0.16.68.0-7a79f4e
-aliyun
        command:
        - sh
        - -c
        - /usr/local/ilogtail/run_logtail.sh 10
        livenessProbe:
          exec:
            command:
            - /etc/init.d/ilogtaild
            - status
          initialDelaySeconds: 30
          periodSeconds: 30
        resources:
          limits:
            memory: 512Mi
          requests:
            cpu: 10m
            memory: 30Mi
        env:
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "${your_aliyun_user_id}"
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "${your_machine_group_user_defined_id} "
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/${your_region_config}/ilogtail_config.json"
          - name: "ALIYUN_LOG_ENV_TAGS"
            value: "_pod_name_|_pod_ip_|_namespace_|_node_name_|_node_ip_"
          - name: "_pod_name_"
            valueFrom:
              fieldRef:
                fieldPath: metadata.name
          - name: "_pod_ip_"
            valueFrom:
              fieldRef:
                fieldPath: status.podIP
          - name: "_namespace_"
            valueFrom:
              fieldRef:
                fieldPath: metadata.namespace
          - name: "_node_name_"
            valueFrom:
              fieldRef:
                fieldPath: spec.nodeName
          - name: "_node_ip_"
            valueFrom:
              fieldRef:
                fieldPath: status.hostIP
```

Parameter	Description
${your_region_config}	Specify a value based on the ID of the region where your Log Service project resides and the type of the network for your project. For more information about regions, see Region names for Logtail installation. ◦ If your project is accessible over the Internet, specify the value in the `region-internet` format. For example, if your project resides in the **China (Hangzhou)** region, specify **cn-hangzhou-Internet**. ◦ If your project is accessible over an internal network of Alibaba Cloud, specify the value in the `region` format. For example, if your project resides in the **China (Hangzhou)** region, specify **cn-hangzhou**.
${your_aliyun_user_id}	Enter the ID of your Alibaba Cloud account. For more information, see Step 1: Obtain the ID of the Alibaba Cloud account for which Log Service is activated.
${your_machine_group_user_defined_id}	Enter the custom identifier that you specify when you create the machine group. **Notice** Make sure that the custom identifier is unique in the region of the Log Service project to which the Logstore belongs.

2. Redeploy the container where Logtail runs.

Step 2: Create a Logtail configuration

1.
2. In the **Import Data** section, click **SkyWalking**.
3. In the **Specify Logstore** step, select the project and Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, click **Use Existing Machine Groups**.
5. In the **Machine Group Settings** step, select the machine group that you want to use in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.
6. In the **Specify Data Source** step, configure the Logtail plug-ins and click **Next**.

   A configuration template is provided in the **Plug-in Config** field. You must replace *${instance}* with the ID of your trace instance.

```
{
      "inputs" : [
        {
            "detail" : {
                "Address" : "0.0.0.0:11800"
            },
            "type" : "service_skywalking_agent_v3"
        }
    ],
    "aggregators" : [
        {
            "detail" : {
                "MetricsLogstore" : "${instance}-metrics",
                "TraceLogstore" : "${instance}-traces"
            },
            "type" : "aggregator_skywalking"
        }
    ],
    "global" : {
        "AlwaysOnline" : true,
        "DelayStopSec" : 300
    }
}
```

Click **Next** to complete the Logtail configuration. Then, Log Service starts to collect logs.

> **Note**
>
> - A Logtail configuration requires a maximum of 3 minutes to take effect.
> - If an error occurs when a Logtail configuration is used to collect logs, follow the instructions provided in How do I view Logtail collection errors? to troubleshoot the error.

FAQ

How do I check whether a Logtail configuration is in effect?

In the container where Logtail runs, run the `cat /usr/local/ilogtail/user_log_config.json | grep skywalking` command.

- If the command output includes **SkyWalking**, the Logtail configuration is in effect.
- If the command output does not include **SkyWalking**, the Logtail configuration is not in effect. In this case, check whether the custom identifier that you specify for the machine group when you deploy the data collection image is the same as the custom identifier that you specify when you create the machine group.

What's next

After you complete the preceding configurations, you can query and analyze the trace data of Apache SkyWalking in the specified Logstore. For more information, see Query and analyze trace data.

3.8.4. Import Ingress trace data from Kubernetes clusters to Log Service

This topic describes how to import Ingress trace data from Kubernetes clusters to the Trace application of Log Service by using OpenTelemetry.

Prerequisites

A trace instance is created. For more information, see Create a trace instance.

Step 1: Install the OpenTelemetry Collector

1. Log on to your Kubernetes cluster.
2. Install cert-manager.

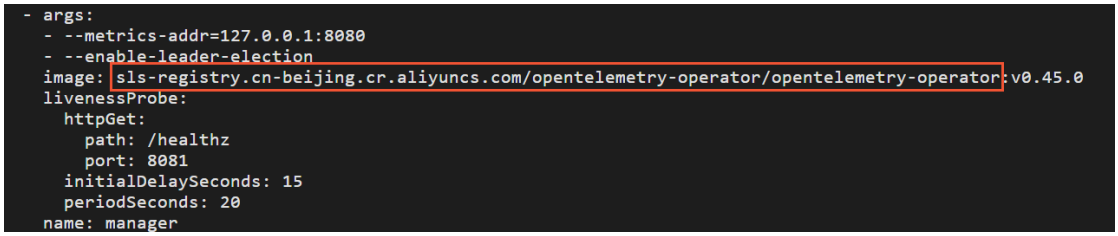
   ```
   kubectl apply -f https://github.com/jetstack/cert-manager/releases/download/v1.6.1/cert-manager.yaml
   ```

3. Deploy the OpenTelemetry Operator.
   i. Download the opentelemetry-operator.yaml file.

   ```
   wget https://github.com/open-telemetry/opentelemetry-operator/releases/latest/download/opentelemetry-operator.yaml
   ```

   ii. Open the opentelemetry-operator.yaml file and replace the image information in the file.

   Replace `ghcr.io/open-telemetry/opentelemetry-operator/opentelemetry-operator` in the opentelemetry-operator.yaml file with `sls-registry.cn-beijing.cr.aliyuncs.com/opentelemetry-operator/opentelemetry-operator`, as shown in the following figure.

   iii. Run the following command to apply the configuration:

   ```
   kubectl apply -f opentelemetry-operator.yaml
   ```

4. Deploy the OpenTelemetry Collector.
   i. Create a YAML file.

   ```
   vim collector.yaml
   ```

   ii. Enter the following code in the YAML file and configure the parameters based on your business scenario:

```
apiVersion: opentelemetry.io/v1alpha1
kind: OpenTelemetryCollector
metadata:
  name: otel
spec:
  image: otel/opentelemetry-collector-contrib:latest
  config: |
    receivers:
      otlp:
        protocols:
          grpc:
          http:
      jaeger:
            protocols:
              grpc:
          thrift_http:
          thrift_compact:
          thrift_binary:
      zipkin:
    exporters:
      alibabacloud_logservice/logs:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-logs"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
      alibabacloud_logservice/metrics:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-traces-metrics"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
      alibabacloud_logservice/traces:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-traces"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
    service:
      pipelines:
        traces:
          receivers: [otlp, jaeger, zipkin]
          exporters: [alibabacloud_logservice/traces]
        metrics:
          receivers: [otlp]
          exporters: [alibabacloud_logservice/metrics]
```

Parameter	Description
endpoint	The Log Service endpoint. Example: cn-hangzhou.log.aliyuncs.com. For more information, see Endpoints.

Parameter	Description
project	The name of the project that you specify when you create a trace instance. For more information, see Create a trace instance.
logstore	The name of the Logstore. After you create a trace instance, Log Service automatically generates three Logstores in the specified project to store log data, metric data, and trace data. Replace the Logstore name based on your business scenario. ■ *trace_instance_id*-logs ■ *trace_instance_id*-traces-metrics ■ *trace_instance_id*-traces *trace_instance_id* specifies the ID of the trace instance. For more information, see Create a trace instance.
access_key_id	The AccessKey ID that is used to access Log Service. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. For more information about how to grant the write permissions on a specified project to a RAM user, see Use custom policies to grant permissions to a RAM user. For more information about how to obtain an AccessKey pair, see AccessKey pair.
access_key_secret	The AccessKey secret that is used to access Log Service. We recommend that you use the AccessKey pair of a RAM user that has only the write permissions on the Log Service project.

iii. Run the following command to apply the configuration.

`otel-test` indicates the namespace where your service resides.

```
kubectl apply -f collector.yaml --namespace=otel-test
```

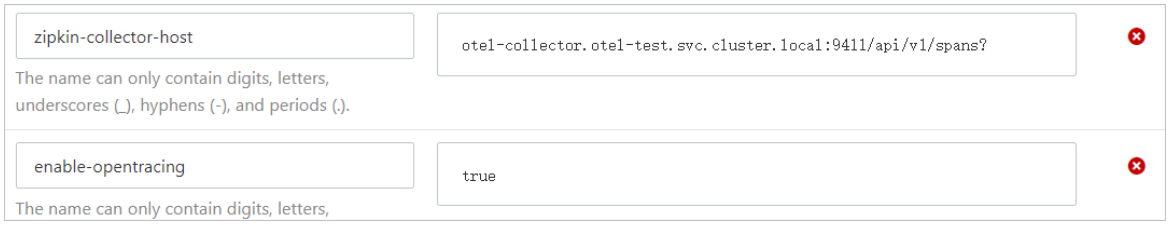
Step 2: Configure Ingress OpenTracing

In this example, an Alibaba Cloud Container Service for Kubernetes (ACK) cluster is used.

1. Log on to the ACK console.
2. On the **Clusters** page, click the cluster that you want to manage.
3. In the left-side navigation pane, choose **Configurations > ConfigMaps**.
4. On the **ConfigMap** page, select **kube-system** from the Namespace drop-down list. In the ConfigMap list, find **nginx-configuration** and click **Edit** in the Actions column.
5. In the **Edit** panel, configure the following two parameters and click **OK**.

   `otel-test` indicates the namespace where your service resides. This namespace must be the same as the namespace that you specified in Step 4.iii.

```
zipkin-collector-host: otel-collector.otel-test.svc.cluster.local:9411/api/v1/spans?
enable-opentracing: true
```



After you complete the preceding configuration, OpenTelemetry uploads the Ingress trace data that is generated by your Kubernetes cluster to Trace. You can view the trace data in the Trace application. For more information, see View the details of a trace.

3.9. FAQ

3.9.1. How do I implement OpenTelemetry automatic instrumentation in a Kubernetes cluster?

This topic describes how to implement OpenTelemetry automatic instrumentation in a Kubernetes cluster to upload trace data to Log Service.

Procedure

1. Log on to your Kubernetes cluster.
2. Install cert-manager.

   ```
   kubectl apply -f https://github.com/jetstack/cert-manager/releases/download/v1.6.1/cert-manager.yaml
   ```

3. Deploy OpenTelemetry Operator.
   i. Download the opentelemetry-operator.yaml file.

   ```
   wget https://github.com/open-telemetry/opentelemetry-operator/releases/latest/download/opentelemetry-operator.yaml
   ```

   ii. Open the opentelemetry-operator.yaml file and replace the image information in the file.

   Replace `ghcr.io/open-telemetry/opentelemetry-operator/opentelemetry-operator` in the opentelemetry-operator.yaml file with `sls-registry.cn-beijing.cr.aliyuncs.com/opentelemetry-operator/opentelemetry-operator`.

   ```
   - args:
     - --metrics-addr=127.0.0.1:8080
     - --enable-leader-election
     image: sls-registry.cn-beijing.cr.aliyuncs.com/opentelemetry-operator/opentelemetry-operator:v0.45.0
     livenessProbe:
       httpGet:
         path: /healthz
         port: 8081
       initialDelaySeconds: 15
       periodSeconds: 20
     name: manager
   ```

   iii. Deploy OpenTelemetry Operator.

   ```
   kubectl apply -f opentelemetry-operator.yaml
   ```

4. Deploy OpenTelemetry Collector.
   i. Create a YAML file.

```
vim collector.yaml
```

   ii. Enter the following script in the YAML file and configure the parameters based on your business requirements:

```
apiVersion: opentelemetry.io/v1alpha1
kind: OpenTelemetryCollector
metadata:
  name: otel
spec:
  image: otel/opentelemetry-collector-contrib:latest
  config:
    receivers:
      otlp:
        protocols:
          grpc:
          http:
    exporters:
      alibabacloud_logservice/logs:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-logs"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
      alibabacloud_logservice/metrics:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-traces-metrics"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
      alibabacloud_logservice/traces:
        endpoint: "cn-hangzhou.log.aliyuncs.com"
        project: "demo-project"
        logstore: "store-traces"
        access_key_id: "access-key-id"
        access_key_secret: "access-key-secret"
    service:
      pipelines:
        traces:
          receivers: [otlp]
          exporters: [alibabacloud_logservice/traces]
        logs:
          receivers: [otlp]
          exporters: [alibabacloud_logservice/logs]
        metrics:
          receivers: [otlp]
          exporters: [alibabacloud_logservice/metrics]
```

Parameter	Description
endpoint	The Log Service endpoint. Example: cn-hangzhou.log.aliyuncs.com. For more information, see Endpoints.
project	The name of the project that you specify when you create a trace instance. For more information, see Create a trace instance.
logstore	The name of the Logstore. After you create a trace instance, Log Service generates three Logstores in the specified project to store log data, metric data, and trace data. Specify the Logstore name based on your business requirements. ■ *trace_instance_id*-logs ■ *trace_instance_id*-traces-metrics ■ *trace_instance_id*-traces *trace_instance_id* specifies the ID of the trace instance. You can replace trace_instance_id based on your business requirements. For more information, see Create a trace instance.
access_key_id	The AccessKey ID that is used to access Log Service. For more information, see AccessKey pair.
access_key_secret	The AccessKey secret that is used to access Log Service. For more information, see AccessKey pair.

iii. Run the following command to apply the configuration:

```
kubectl apply -f collector.yaml
```

5. Deploy OpenTelemetry Auto-Instrumentation.

i. Create a YAML file.

```
vim instrumentation.yaml
```

ii. Enter the following script in the YAML file:

```
apiVersion: opentelemetry.io/v1alpha1
kind: Instrumentation
metadata:
  name: my-instrumentation
spec:
  exporter:
    endpoint: http://otel-collector:4317
  propagators:
    - tracecontext
    - baggage
    - b3
  java:
    image: ghcr.io/open-telemetry/opentelemetry-operator/autoinstrumentation-java:l
atest
  nodejs:
    image: ghcr.io/open-telemetry/opentelemetry-operator/autoinstrumentation-nodejs
:latest
  python:
    image: ghcr.io/open-telemetry/opentelemetry-operator/autoinstrumentation-python
:latest
```

iii. Configure an environment variable to pass the service name and module name to the container.

```
OTEL_RESOURCE_ATTRIBUTES=service.name=your_service,service.namespace=your_service_n
amespace
```

Parameter	Description
service.name	The name of the service. Replace your_service with the actual value.
service.namespace	The name of the module. Replace your_service_namespace with the actual value.

iv. Run the following command to apply the configuration:

```
kubectl apply -f instrumentation.yaml
```

6. Add the configuration for automatic instrumentation to your configuration file.

   Add the configuration script to the configuration file of your application based on your business requirements. Only Python, Node.js, and Java applications are supported.

- Java

  ```
  instrumentation.opentelemetry.io/inject-java: 'true'
  ```

- Python

  ```
  instrumentation.opentelemetry.io/inject-python: "true"
  ```

- Node.js

  ```
  instrumentation.opentelemetry.io/inject-nodejs: "true"
  ```

4.Full-stack Monitoring

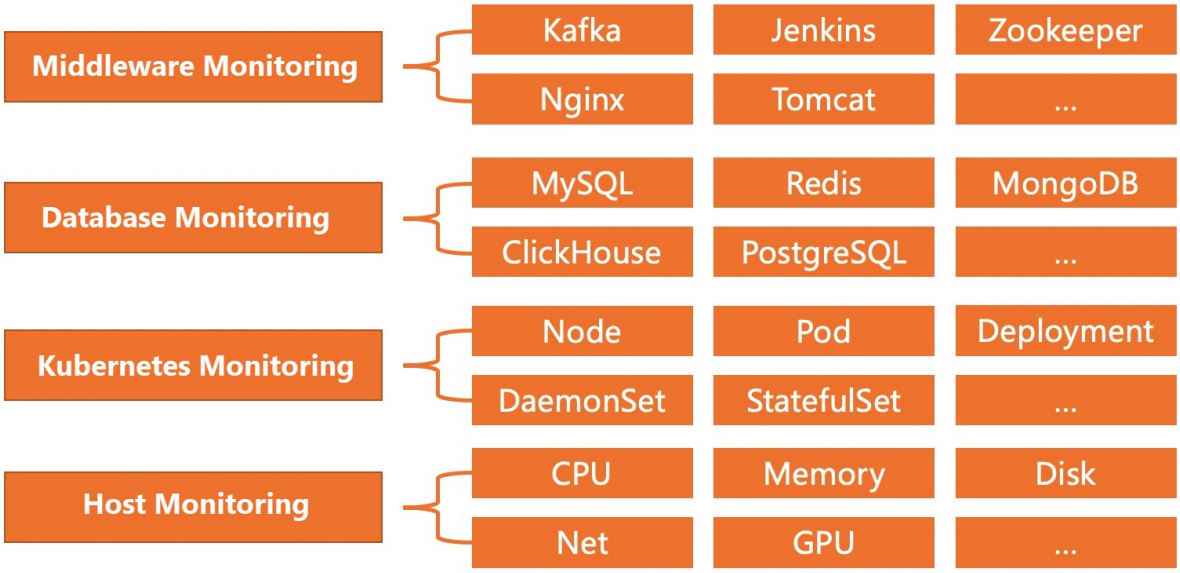
4.1. Overview of Full-stack Monitoring

Log Service provides the Full-stack Monitoring application to monitor IT systems from end to end. The application can monitor various system components, such as hosts, Kubernetes clusters, databases, and middleware. This topic describes the features, benefits, assets, and billing for Full-stack Monitoring.

Features

Full-stack Monitoring is based on Log Service capabilities, such as collection, storage, analysis, visualization, alerting, and AIOps. Full-stack Monitoring provides the following features:

- Allows you to install Logtail on an Elastic Compute Service (ECS) instance or a Kubernetes cluster with a few clicks, manage monitoring configurations in a visualized manner, and configure metrics without the need to log on to hosts.
- Provides visualized built-in reports, such as resource overview, resource usage monitoring, hotspot analysis, and detailed metrics.
- Supports custom analysis and different languages for analysis, such as PromQL and SQL-92.
- Supports AIOps metric inspection and detects abnormal metrics by using machine learning.
- Supports custom alert settings and sends alert notifications by using the following methods: Message Center, text messages, emails, voice calls, DingTalk, and custom webhooks.
- Monitors various system components in real time, such as hosts, Kubernetes clusters, databases, and middleware.



Benefits

- Simple: You can enable Full-stack Monitoring with a few clicks and use centralized storage for Full-stack Monitoring. Full-stack Monitoring provides built-in reports to meet different requirements, such as monitoring dashboards, troubleshooting, and convergence and analysis of observability data.
- Massive: Metric storage is supported by the time series storage engine of Log Service. You can write and query time series data at an ultra large scale.

- Real-time: Real-time monitoring data is required in multiple scenarios, such as DevOps, monitoring, and alerting. Full-stack Monitoring allows you to specify collection intervals as small as 5 seconds.
- Elastic: You can filter metrics or specify a custom retention period for each metric. The storage of a Metricstore can be dynamically scaled to meet service requirements.
- Intelligent: Full-stack Monitoring can identify and locate errors in an efficient and accurate manner because it uses the stream inspection algorithm that is developed by Alibaba DAMO Academy for AIOps.

Assets

After specified data is collected to Log Service, Log Service automatically creates the following assets in a specified project: Logstores, Metricstores, and dashboards. {instance} specifies the ID of the Full-stack Monitoring instance that you create.

Monitoring type	Logstore	Metricstore	Dashboard
Host monitoring	*{instance}*-metas: stores the configuration data of hosts. The data includes CPU models and memory sizes.	*{instance}*-node-metrics: stores the metric data of hosts. The data includes CPU utilization and memory usage.	• Resource Overview • Hosts • Hotspot Analysis • Standalone Metrics-Simple • Standalone Metrics-Details

Kubernetes monitoring	*{instance}*-metas: stores the configuration data of Kubernetes nodes.	• *{instance}*-node-metrics: stores the metric data of Kubernetes nodes. The data includes CPU utilization and memory usage.
		• *{instance}*-k8s-metrics: stores the metric data of Kubernetes clusters. The data includes pod, node, and Deployment metrics.
			• Resource Overview
			• Water-level Monitoring
			• Run-time Monitoring
			• Core Component Monitoring
			• Resource Monitoring
			  ◦ Nodes
			  ◦ Node Metrics
			  ◦ Pods
			  ◦ Pod Metrics
			  ◦ Deployments
			  ◦ Deployment Metrics
			  ◦ StatefulSets
			  ◦ StatefulSet Metrics
			  ◦ DaemonSets
			  ◦ DaemonSet Metrics

Monitoring type	Logstore	Metricstore	Dashboard
Middleware monitoring	None.	*{instance}*-common-metrics: stores the metric data of middleware.	• JVM Monitoring • NGINX Monitoring • Tomcat Monitoring • Kafka Monitoring • Nvidia GPU Monitoring
Database monitoring	None.	*{instance}*-common-metrics: stores the metric data of databases.	• MySQL Monitoring • Redis Monitoring • ClickHouse Monitoring • Elasticsearch Monitoring • MongoDB Monitoring

Billing

You can use Full-stack Monitoring free of charge. However, you are charged for operations related to Full-stack Monitoring, such as storage and indexing. For more information, see Billable items.

4.2. Create an instance

A Full-stack Monitoring instance is used to manage items such as monitoring data and dashboards. This topic describes how to create an instance.

Prerequisites

A project is created. For more information, see Create a project.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click **Add**.
4. In the **Create Instance** panel, configure the following parameters and click **OK**.

Parameter	Description
Instance ID	The ID of the instance.
Instance Name	The name of the instance.
Project	The project that you want to use. After monitoring data is collected to Log Service, Log Service automatically creates the required assets for the data in the project, such as Logstores, Metricstores, and dashboards.
Region	The region where the project resides.

What's next

- Collect monitoring data from hosts
- Collect monitoring data from Kubernetes clusters
- Collect monitoring data from Kafka servers
- Collect monitoring data from NGINX
- Collect monitoring data from NVIDIA GPU servers
- Collect monitoring data from Tomcat servers
- Collect monitoring data from JVM servers
- Collect monitoring data from MySQL databases
- Collect monitoring data from Redis databases
- Collect monitoring data from Elasticsearch clusters
- Collect monitoring data from ClickHouse databases
- Collect monitoring data from MongoDB databases

4.3. Collect data to Log Service

4.3.1. Collect monitoring data from hosts

You can collect the configuration data and metric data of hosts to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner. The configuration data includes CPU models and memory sizes. The metric data includes CPU utilization and memory usage.

Prerequisites

An instance is created. For more information, see Create an instance.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Host.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the  icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.40 or later on the server. For more information, see Install Logtail on a Linux server.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.

7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

   Log Service provides the following two collection plug-ins for host monitoring:

   - metric_system_v2: used to collect the metric data of hosts. The data includes CPU utilization and memory usage. The collected data is stored in a Metricstore named {instance}-node-metrics.
   - metric_meta_host: used to collect the configuration data of hosts. The data includes CPU models and memory sizes. The collected data is stored in a Logstore named {instance}-metas.

   {instance} specifies the ID of the Full-stack Monitoring instance that you create.

Parameter	Description	Example
Config Name	The name of the Logtail configuration.	host-szytbsxv

node-metrics	The configurations of the metric_system_v2 plug-in that is used to collect the metric data of hosts. The data includes CPU utilization and memory usage.
	- IntervalMs: the interval of requests. Unit: ms. The value must be greater than or equal to 5000. We recommend that you set the value to 30000.
	- cluster: the name of the cluster. The name must meet the following requirements:
	  ■ The name can contain lowercase letters, digits, hyphens (-), and underscores (_).
	  ■ The name must start and end with a lowercase letter or a digit.
	  ■ The name must be 3 to 63 characters in length.
	After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the host monitoring data that is collected by using the Logtail configuration.
	- type: the type of the data source. Set the value to metric_system_v2.
		<pre>{
		  "inputs": [
		    {
		      "detail": {
		        "IntervalMs": 30000,
		        "Labels": {
		          "cluster": "my-cluster"
		        }
		      },
		      "type": "metric_system_v2"
		    }
		  ]
		}</pre>

Parameter	Description	Example

node-metas	The configurations of the metric_meta_host plug-in that is used to collect the configuration data of hosts. The data includes CPU models and memory sizes.
	○ IntervalMs: the interval of requests. Unit: ms. The value must be greater than or equal to 5000. We recommend that you set the value to 30000.
	○ cluster: the name of the cluster. The name must meet the following requirements:
	■ The name can contain lowercase letters, digits, hyphens (-), and underscores (_).
	■ The name must start and end with a lowercase letter or a digit.
	■ The name must be 3 to 63 characters in length.
	After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the host monitoring data that is collected by using the Logtail configuration.
	○ type: the type of the data source. Set the value to metric_meta_host.
		<pre>{
		  "inputs": [
		    {
		      "detail": {
		        "IntervalMs": 30000,
		        "Labels": {
		          "cluster": "my-cluster"
		        }
		      },
		      "type": "metric_meta_host"
		    }
		  ]
		}</pre>

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After host monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.2. Collect monitoring data from Kubernetes clusters

Kubernetes provides multiple metrics. You can collect the metrics from Kubernetes clusters to the Full-stack Monitoring application. This way, you can monitor data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- Logtail components are installed in a Kubernetes cluster. For more information, see Install Logtail

components.

> **Note** Make sure that you select the same project when you install Logtail components and when you create a Full-stack Monitoring instance.

Install a monitoring component

1. Log on to your Kubernetes cluster.
2. Download the installation script sls-monitoring.sh.
3. Run the following command to install the monitoring component:

```
bash sls-monitoring.sh --project ${project} --aliuid ${aliuid} --instance-id ${instance
-id} --cluster-id ${cluster-id} --action install --config-params ${config-params} --kub
e-state-metrics-image-name ${kube-state-metrics-image} --logtail-image-name ${logtail-i
mage}
```

Configuration example:

```
bash sls-monitoring.sh --project k8s-log-c0ae5df1 --aliuid 165****50  --instance-id my-
monitor --cluster-id my-k8s --action install --config-params cn-beijing --kube-state-me
trics-image-name registry.cn-beijing.aliyuncs.com/log-service/kube-state-metrics --logt
ail-image-name registry.cn-beijing.aliyuncs.com/log-service/logtail
```

The following table describes the parameters that are included in the command. You can configure the parameters based on your business requirements.

Parameter	Required	Description
${project}	Yes	The name of the Log Service project. You must enter the name of the project that you select when you create the Full-stack Monitoring instance. For more information, see Create an instance.
${aliuid}	Yes	The ID of your Alibaba Cloud account. For more information, see Step 1: Obtain the ID of the Alibaba Cloud account for which Log Service is activated.
${instance-id}	Yes	The ID of the Full-stack Monitoring instance.
${cluster-id}	Yes	The ID of your Kubernetes cluster. The ID must meet the following requirements: ○ The ID can contain lowercase letters, digits, hyphens (-), and underscores (_). ○ The ID can be up to 24 characters in length.

Parameter	Required	Description
${config-params}	Yes	Specify a value based on the region where your Log Service project resides and the type of the network for your project. For more information about regions, see Region names for Logtail installation. ○ If data is collected to your project over the Internet, specify the value in the `region-internet` format. For example, if your project resides in the **China (Hangzhou)** region, specify **cn-hangzhou-Internet**. ○ If data is collected to your project over an internal network of Alibaba Cloud, specify the value in the `region` format. For example, if your project resides in the **China (Hangzhou)** region, specify **cn-hangzhou**.
${kube-state-metrics-image}	Yes	The endpoint of the image for kube-state-metrics. Default value: registry.cn-beijing.aliyuncs.com/log-service/kube-state-metrics. **Notice** The default value is a public endpoint. If your cluster cannot access the Internet, you can enter an internal endpoint based on the region of the image. Example: registry-vpc.cn-hangzhou.aliyuncs.com/log-service/kube-state-metrics.
${logtail-image}	Yes	The endpoint of the image for Logtail. Default value: registry.cn-beijing.aliyuncs.com/log-service/logtail. **Notice** The default value is a public endpoint. If your cluster cannot access the Internet, you can enter an internal endpoint based on the region of the image. Example: registry-vpc.cn-hangzhou.aliyuncs.com/log-service/logtail.
kube-state-metrics-image-tag	No	The label of the image for kube-state-metrics. Example: v1.6.0-f4ec1f70-aliyun. **Note** The default value varies based on the version of the release. You can view the default value in the sls-monitoring.sh script.

Parameter	Required	Description
logtail-image-tag	No	The label of the image for Logtail. Example: v1.0.27.0-7052198-aliyun. **Note** The default value varies based on the version of the release. You can view the default value in the sls-monitoring.sh script.
logtail-cluster-limit-cpu	No	The CPU limit of the Logtail cluster. Example: 4000 m.
logtail-cluster-limit-mem	No	The memory limit of the Logtail cluster. Example: 4096 Mi.
logtail-nodeds-limit-cpu	No	The CPU limit of a Logtail node. Example: 200 m.
logtail-nodeds-limit-mem	No	The memory limit of a Logtail node. Example: 384 Mi.
kube-state-metrics-limit-cpu	No	The CPU limit of kube-state-metrics. Example: 1000 m.
kube-state-metrics-limit-mem	No	The memory limit of kube-state-metrics. Example: 1024 Mi.

After the sls-monitoring.sh script is run, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

4. Confirm the created Logtail configurations.
   i.
   ii. In the Projects list, click the project that you specify in Step .
   iii. View Logtail configurations in the Logstores and Metricstores that are used.

   After the sls-monitoring.sh script is run, Log Service automatically creates four Logtail configurations. You can modify the configurations in the AliyunLogConfig Custom Resource Definitions (CRDs) in the sls-monitoring namespace. You can run the `kubectl get aliyunlogconfig -n sls-monitoring` command to query the CRDs.

Logstore and Metricstore	Logtail configuration
{instance}-metas Logstore	■ {instance}__node-metas__{cluster}: used to collect configuration data from Kubernetes nodes. ■ {instance}__k8s-metas__{cluster}: used to collect configuration data from Kubernetes clusters.
{instance}-node-metrics Metricstore	{instance}__node-metrics__{cluster}: used to collect metric data from Kubernetes nodes.
{instance}-k8s-metrics Metricstore	{instance}__k8s-metrics__{cluster}: used to collect metric data from Kubernetes clusters.

5. Log on to your Kubernetes cluster and confirm that the pods in the sls-monitoring namespace are all in the Running state.

Resources for the monitoring component

The Kubernetes resources that are used to collect Kubernetes monitoring data are all created in the sls-monitoring namespace. The resources include two Deployments, one DaemonSet, and four AliyunLogConfig CRDs.

Resource type	Resource name	Description
Deployment	kube-state-metrics	Used to listen to Kubernetes API operations to obtain the metric data of Kubernetes clusters.
	logtail-kubernetes-metrics	Used to collect metric data from the kube-state-metrics container to Log Service.
DaemonSet	logtail-node-monitoring-ds	Used to deploy the Logtail container in automatic mode. By default, Logtail runs on each node and collects the configuration data and metric data of the nodes.
AliyunLogConfig	{instance-id}-k8s-metas	Used to collect the configuration data of Kubernetes clusters. The data includes the name, namespace, label, image, and limit of each Deployment, pod, Ingress, and Service. By default, the collected data is stored in a Logstore named {instance}-metas.
	{instance-id}-k8s-metrics	Used to collect the metric data of Kubernetes clusters. The data includes the CPU, memory, and network data of pods and containers. By default, the collected data is stored in a Metricstore named {instance}-k8s-metrics.
	{instance-id}-node-metas	Used to collect the configuration data of Kubernetes nodes. The data includes CPU models and memory sizes. By default, the collected data is stored in a Logstore named {instance}-metas.
	{instance-id}-node-metrics	Used to collect the metric data of Kubernetes nodes. The data includes CPU utilization and memory usage. By default, the collected data is stored in a Metricstore named {instance}-node-metrics.

What's next

After Kubernetes monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.3. Collect monitoring data from middleware

4.3.3.1. Collect monitoring data from Kafka servers

Kafka servers support multiple metrics. You can collect metric data from Kafka servers to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- Java 1.6 or later is installed on your Kafka server.

Step 1: Create a Logtail configuration

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Kafka.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the Kafka server whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the Kafka cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the Kafka monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the Kafka server. The information includes the following configuration items: ○ **Address**: the address of the Kafka server. You can enter the IP address, hostname, or domain name of the server. ○ **Port**: the port number of the Kafka server. Default value: 7777. You can add information about multiple Kafka servers based on your business requirements.
Custom Tags	The custom tags that are added to the collected Kafka monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the Kafka monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

Step 2: Configure JavaAgent

After you create the Logtail configuration, you must configure JavaAgent on the Kafka server. Log Service allows you to use Jolokia to configure JavaAgent. For more information, see Jolokia. You can download and use Jolokia based on the official documentation of Jolokia. You can also use Jolokia JavaAgent that is provided together with Logtail in Log Service. Jolokia JavaAgent is stored in `/etc/logtail/telegraf/javaagent/jolokia-jvm.jar`.

1. Configure the environment variable `KAFKA_JVM_PERFORMANCE_OPTS` for the Kafka server.

   For example, specify `export KAFKA_JVM_PERFORMANCE_OPTS=-javaagent:/etc/logtail/telegraf/javaagent/jolokia-jvm.jar=port=7777`. 7777 is the port number of the server. Make sure that the port number is the same as the port number that you specify in Step 1: Create a Logtail configuration.

> **Note** By default, Jolokia JavaAgent listens only on the IP address 127.0.0.1 and allows requests only from the local host. If Logtail and your Java application are installed on different servers, you can add the host= field to the added script. This way, Jolokia JavaAgent can listen on other IP addresses. If you add **host=0.0.0.0**, Jolokia JavaAgent listens on all IP addresses. Example:
>
> ```
> -javaagent:/tmp/jolokia-jvm.jar=port=7777,host=0.0.0.0
> ```

2. Restart your Java application.

   If your Java application fails to restart, run the following command to connect Jolokia JavaAgent to a specified Java process. This way, the configuration immediately takes effect. Replace Java PID with the actual value.

   > **Notice** This operation is used only for testing. In actual scenarios, you must complete the configuration based on preceding descriptions. Otherwise, the configuration becomes invalid after your application restarts.

   ```
   java -jar /etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar --port 7777 start Java PID
   ```

   If information similar to the following code is returned, the connection is successful:

   ```
   Jolokia is already attached to PID 752
   http://127.0.0.1:7777/jolokia/
   ```

3. Access the following URL to verify the connection:

   ```
   curl http://127.0.0.1:7777/jolokia/
   ```

   If information similar to the following code is returned, the connection is normal:

   ```
   {"request":{"type":"version"},"value":{"agent":"1.6.2","protocol":"7.2","config":{"list
   enForHttpService":"true","maxCollectionSize":"0","authIgnoreCerts":"false","agentId":"3
   0.43.124.186-752-5b091b5d-jvm","debug":"false","agentType":"jvm","policyLocation":"clas
   spath:\/jolokia-access.xml","agentContext":"\/jolokia","serializeException":"false","mi
   meType":"text\/plain","maxDepth":"15","authMode":"basic","authMatch":"any","discoveryEn
   abled":"true","streaming":"true","canonicalNaming":"true","historyMaxEntries":"10","all
   owErrorDetails":"true","allowDnsReverseLookup":"true","realm":"jolokia","includeStackTr
   ace":"true","maxObjects":"0","useRestrictorService":"false","debugMaxEntries":"100"},"i
   nfo":{"product":"tomcat","vendor":"Apache","version":"8.5.57"}},"timestamp":1602663330,
   "status":200}
   ```

What's next

After Kafka monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.3.2. Collect monitoring data from NGINX

NGINX comes with a status page that reports metrics about NGINX. You can collect metric data from NGINX servers to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- The NGINX status module is configured. For more information, see Configure the NGINX status module.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable NGINX.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.50 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the NGINX server whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the NGINX cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the NGINX monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the NGINX server. The information includes the following configuration items: ○ **Address**: the address of the NGINX server. You can enter the IP address, hostname, or domain name of the server. ○ **Port**: the port number of the NGINX server. ○ **Path**: the URI of the NGINX status module. Example: */private/nginx_status*. For more information about how to configure the NGINX status module, see Configure the NGINX status module. You can add information about multiple NGINX servers based on your business requirements.
Custom Tags	The custom tags that are added to the collected NGINX monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the NGINX monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After NGINX monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.3.3. Collect monitoring data from NVIDIA GPU servers

NVIDIA GPU servers support multiple metrics. You can collect metric data from NVIDIA GPU servers to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- An NVIDIA GPU driver is installed. For more information, see Install an NVIDIA GPU driver.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Nvidia GPU.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.50 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the NVIDIA GPU server whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the NVIDIA GPU cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the NVIDIA GPU monitoring data that is collected by using the Logtail configuration. > **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.

Parameter	Description
Nvidia SMI Path	The directory in which nvidia-smi is installed. Default value: */usr/bin/nvidia-smi*.
Custom Tags	The custom tags that are added to the collected NVIDIA GPU monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the NVIDIA GPU monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After NVIDIA GPU monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.3.4. Collect monitoring data from Tomcat servers

Tomcat servers support multiple metrics. You can collect metric data from Tomcat servers to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- Java 1.6 or later is installed on your server.

Step 1: Create a Logtail configuration

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Tomcat.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

> **Notice** Make sure that the server on which you want to install Logtail can connect to the Tomcat server whose metric data you want to collect.

6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.

7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the Tomcat cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the Tomcat monitoring data that is collected by using the Logtail configuration. > **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Application Name	The name of the Java application. You can enter a custom value.
Server List	The information about the Tomcat server. The information includes the following configuration items: ○ **Address**: the address of the Tomcat server. ○ **Port**: the port number of the Tomcat server. Default value: 7777. You can add information about multiple Tomcat servers based on your business requirements.
Custom Tags	The custom tags that are added to the collected Tomcat monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the Tomcat monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

Step 2: Configure JavaAgent

After you create the Logtail configuration, you must configure JavaAgent on the Tomcat server. Log Service allows you to use Jolokia to configure JavaAgent. For more information, see Jolokia. You can download and use Jolokia based on the official documentation of Jolokia. You can also use Jolokia JavaAgent that is provided together with Logtail in Log Service. Jolokia JavaAgent is stored in */etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar*.

1. Configure the environment variable `JAVA_OPTS`.

   For example, specify `export JAVA_OPTS="-javaagent:/etc/ilogtail/telegraf/jolokia-jvm.jar=port=7777"`. 7777 is the port number of the Tomcat server. Make sure that the port number is the same as the port number that you specify in Step 1: Create a Logtail configuration.

   > **Note** By default, Jolokia JavaAgent listens only on the IP address 127.0.0.1 and allows requests only from the local host. If Logtail and your Java application are installed on different servers, you can add the host= field to the added script. This way, Jolokia JavaAgent can listen on other IP addresses. If you add **host=0.0.0.0**, Jolokia JavaAgent listens on all IP addresses. Example:
   >
   > ```
   > -javaagent:/tmp/jolokia-jvm.jar=port=7777,host=0.0.0.0
   > ```
   >
   > Replace the path to the jolokia-jvm.jar package with the actual value.

2. Restart your Java application.

   If your Java application fails to restart, run the following command to connect Jolokia JavaAgent to a specified Java process. This way, the configuration immediately takes effect. Replace Java PID with the actual value.

   > **Notice** This operation is used only for testing. In actual scenarios, you must complete the configuration based on preceding descriptions. Otherwise, the configuration becomes invalid after your application restarts.

   ```
   java -jar /etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar --port 7777 start Java PID
   ```

   If information similar to the following code is returned, the connection is successful:

   ```
   Jolokia is already attached to PID 752
   http://127.0.0.1:7777/jolokia/
   ```

3. Access the following URL to verify the connection:

   ```
   curl http://127.0.0.1:7777/jolokia/
   ```

   If information similar to the following code is returned, the connection is normal:

```
{"request":{"type":"version"},"value":{"agent":"1.6.2","protocol":"7.2","config":{"list
enForHttpService":"true","maxCollectionSize":"0","authIgnoreCerts":"false","agentId":"3
0.43.124.186-752-5b091b5d-jvm","debug":"false","agentType":"jvm","policyLocation":"clas
spath:\/jolokia-access.xml","agentContext":"\/jolokia","serializeException":"false","mi
meType":"text\/plain","maxDepth":"15","authMode":"basic","authMatch":"any","discoveryEn
abled":"true","streaming":"true","canonicalNaming":"true","historyMaxEntries":"10","all
owErrorDetails":"true","allowDnsReverseLookup":"true","realm":"jolokia","includeStackTr
ace":"true","maxObjects":"0","useRestrictorService":"false","debugMaxEntries":"100"},"i
nfo":{"product":"tomcat","vendor":"Apache","version":"8.5.57"}},"timestamp":1602663330,
"status":200}
```


What's next

After Tomcat monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.3.5. Collect monitoring data from JVM servers

Java Virtual Machine (JVM) supports multiple metrics. You can collect metric data from JVM servers to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- Java 1.6 or later is installed on your server.

Step 1: Create a Logtail configuration

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable JVM.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the JVM server whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.

7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

> **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the JVM cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the JVM monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Application Name	The name of the Java application. You can enter a custom value.
Server List	The information about the JVM server. The information includes the following configuration items: ○ **Address**: the address of the JVM server. ○ **Port**: the port number of the JVM server. Default value: 7777. You can add information about multiple JVM servers based on your business requirements.
Custom Tags	The custom tags that are added to the collected JVM monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the JVM monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

Step 2: Configure JavaAgent

After you create the Logtail configuration, you must configure JavaAgent on the JVM server. Log Service allows you to use Jolokia to configure JavaAgent. For more information, see Jolokia. You can download and use Jolokia based on the official documentation of Jolokia. You can also use Jolokia JavaAgent that is provided together with Logtail in Log Service. Jolokia JavaAgent is stored in */etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar*.

1. Add the script `-javaagent:/etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar=port=7777` to the Java startup parameters.

   7777 is the port number of the server. Make sure that the port number is the same as the port number that you specify in Step 1: Create a Logtail configuration.

   > **Note** By default, Jolokia JavaAgent listens only on the IP address 127.0.0.1 and allows requests only from the local host. If Logtail and your Java application are installed on different servers, you can add the host= field to the added script. This way, Jolokia JavaAgent can listen on other IP addresses. If you add **host=0.0.0.0**, Jolokia JavaAgent listens on all IP addresses. Example:
   >
   > ```
   > -javaagent:/tmp/jolokia-jvm.jar=port=7777,host=0.0.0.0
   > ```

2. Restart your Java application.

   If your Java application fails to restart, run the following command to connect Jolokia JavaAgent to a specified Java process. This way, the configuration immediately takes effect. Replace Java PID with the actual value.

   > **Notice** This operation is used only for testing. In actual scenarios, you must complete the configuration based on preceding descriptions. Otherwise, the configuration becomes invalid after your application restarts.

   ```
   java -jar /etc/ilogtail/telegraf/javaagent/jolokia-jvm.jar --port 7777 start Java PID
   ```

   If information similar to the following code is returned, the connection is successful:

   ```
   Jolokia is already attached to PID 752
   http://127.0.0.1:7777/jolokia/
   ```

3. Access the following URL to verify the connection:

   ```
   curl http://127.0.0.1:7777/jolokia/
   ```

   If information similar to the following code is returned, the connection is normal:

   ```
   {"request":{"type":"version"},"value":{"agent":"1.6.2","protocol":"7.2","config":{"list
   enForHttpService":"true","maxCollectionSize":"0","authIgnoreCerts":"false","agentId":"3
   0.43.124.186-752-5b091b5d-jvm","debug":"false","agentType":"jvm","policyLocation":"clas
   spath:\/jolokia-access.xml","agentContext":"\/jolokia","serializeException":"false","mi
   meType":"text\/plain","maxDepth":"15","authMode":"basic","authMatch":"any","discoveryEn
   abled":"true","streaming":"true","canonicalNaming":"true","historyMaxEntries":"10","all
   owErrorDetails":"true","allowDnsReverseLookup":"true","realm":"jolokia","includeStackTr
   ace":"true","maxObjects":"0","useRestrictorService":"false","debugMaxEntries":"100"},"i
   nfo":{"product":"tomcat","vendor":"Apache","version":"8.5.57"}},"timestamp":1602663330,
   "status":200}
   ```

What's next

After JVM monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.4. Collect monitoring data from databases

4.3.4.1. Collect monitoring data from MySQL databases

MySQL databases support multiple metrics. You can collect metric data from MySQL databases to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

- An instance is created. For more information, see Create an instance.
- A MySQL database is available.

  Only MySQL 5.5 or later is supported.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable MySQL.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the server on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the MySQL database whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

> **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the MySQL cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the MySQL monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the MySQL database. The information includes the following configuration items: ○ **Account**: the username of the account that you use to connect to the MySQL database. **Note** We recommend that you create a dedicated account to monitor the data of the MySQL database and grant the account only the permissions that are required to monitor data. ○ **Password**: the password of the account that you use to connect to the MySQL database. ○ **Address**: the address of the MySQL database. You can enter the IP address, hostname, or domain name of the server that hosts the database. Example: rm-bp15r****t9v5.mysql.rds.aliyuncs.com. This is the internal endpoint of an ApsaraDB RDS for MySQL instance. ○ **Port**: the port number of the MySQL database. Default value: 3306. You can add information about multiple MySQL databases based on your business requirements.
Custom Tags	The custom tags that are added to the collected MySQL monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the MySQL monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After MySQL monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.4.2. Collect monitoring data from Redis databases

Redis databases support multiple metrics. You can collect metric data from Redis databases to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

An instance is created. For more information, see Create an instance.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Redis.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the server on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the Redis database whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.
Cluster Name	The name of the Redis cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the Redis monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the Redis database. The information includes the following configuration items: ○ **Address**: the address of the Redis database. You can enter the IP address, hostname, or domain name of the server that hosts the database. ○ **Port**: the port number of the Redis database. Default value: 6379. You can add information about multiple Redis databases based on your business requirements.
Password	If authentication is configured for the Redis database, you must enter the password of the Redis database.
Custom Tags	The custom tags that are added to the collected Redis monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the Redis monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After Redis monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.4.3. Collect monitoring data from Elasticsearch clusters

Elasticsearch clusters support multiple metrics. You can collect metric data from Elasticsearch clusters to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

An instance is created. For more information, see Create an instance.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable Elasticsearch.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the  icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the Elasticsearch cluster whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.

Parameter	Description
Cluster Name	The name of the Elasticsearch cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the Elasticsearch monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the Elasticsearch cluster. The information includes the following configuration items: ○ **Address**: the address of the Elasticsearch cluster. You can enter the IP address, hostname, or domain name of the server in the cluster. ○ **Port**: the port number of the Elasticsearch cluster. Default value: 9200. You can add information about multiple Elasticsearch clusters based on your business requirements.
Password	If authentication is configured for the Elasticsearch cluster, you must enter the account and password that you use to connect to the Elasticsearch cluster.
Index Name	The name of the index that is created in the Elasticsearch cluster. If you enter **_all**, the metric data of all indexes in the Elasticsearch cluster is collected.
Custom Tags	The custom tags that are added to the collected Elasticsearch monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the Elasticsearch monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After Elasticsearch monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.4.4. Collect monitoring data from ClickHouse databases

ClickHouse databases support multiple metrics. You can collect metric data from ClickHouse databases to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

An instance is created. For more information, see Create an instance.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable ClickHouse.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.48 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the ClickHouse database whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Specify Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.

Parameter	Description
Cluster Name	The name of the ClickHouse cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the ClickHouse monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Username	The username of the account that you use to connect to the ClickHouse database.
Password	The password of the account that you use to connect to the ClickHouse database.
Server List	The information about the ClickHouse database. The information includes the following configuration items: ○ **Address**: the address of the ClickHouse database. ○ **Port**: the port number of the ClickHouse database. Default value: 8123. You can add information about multiple ClickHouse databases based on your business requirements.
Custom Tags	The custom tags that are added to the collected ClickHouse monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the ClickHouse monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After ClickHouse monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.3.4.5. Collect monitoring data from MongoDB databases

MongoDB databases support multiple metrics. You can collect metric data from MongoDB databases to the Full-stack Monitoring application. This way, you can monitor the data in a visualized manner.

Prerequisites

An instance is created. For more information, see Create an instance.

Procedure

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance.
4. On the **Data Import** page, enable MongoDB.

   If this is your first time to create a Logtail configuration for host monitoring, turn on the switch to go to the configuration page. If you have created a Logtail configuration, click the  icon to go to the configuration page.
5. In the **Install Logtail** step, select the machine on which you want to install Logtail and click **Next**.
   - If you want to install Logtail on an Elastic Compute Service (ECS) instance, select the ECS instance on the **ECS Instances** tab and click **Execute Now**. For more information, see Install Logtail on ECS instances.
   - If you want to install Logtail on a self-managed Linux server or a Linux server from a third-party cloud, you must manually install Logtail V0.16.50 or later on the server. For more information, see Install Logtail on a Linux server.

   > **Notice** Make sure that the server on which you want to install Logtail can connect to the MongoDB database whose metric data you want to collect.
6. In the **Create Machine Group** step, create a machine group and click **Next**.

   Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.
7. In the **Machine Group Settings** step, select the machine group that you create in the **Source Server Groups** section and move the machine group to the **Applied Server Groups** section. Then, click **Next**.

   > **Notice** If you immediately apply a machine group after it is created, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. In this case, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
8. In the **Configure Data Source** step, configure the following parameters and click **Complete**.

Parameter	Description
Configuration Name	The name of the Logtail configuration. You can enter a custom value.

Parameter	Description
Cluster Name	The name of the MongoDB cluster. You can enter a custom value. After you configure this parameter, Log Service adds a *cluster=Cluster name* tag to the MongoDB monitoring data that is collected by using the Logtail configuration. **Notice** Make sure that the cluster name is unique. Otherwise, data conflicts may occur.
Server List	The information about the MongoDB database. ○ **Address**: the address of the MongoDB database. You can enter the IP address, hostname, or domain name of the server that hosts the database. ○ **Port**: the port number of the MongoDB database. Default value: 3717. ○ **Account**: the username of the account that you use to connect to the MongoDB database. **Note** We recommend that you create a dedicated account to monitor the data of the MongoDB database and grant the account only the permissions that are required to monitor data. ○ **Password**: the password of the account that you use to connect to the MongoDB database. You can add information about multiple MongoDB databases based on your business requirements.
Custom Tags	The custom tags that are added to the collected MongoDB monitoring data. The tags are key-value pairs. After you configure this parameter, Log Service adds the custom tags to the MongoDB monitoring data that is collected by using the Logtail configuration.

After you complete the configurations, Log Service automatically creates assets such as Metricstores. For more information, see Assets.

What's next

After MongoDB monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dedicated dashboards for the monitoring data. You can use the dashboards to analyze the monitoring data. For more information, see View dashboards.

4.4. View dashboards

After monitoring data is collected to Log Service, the Full-stack Monitoring application automatically creates dashboards to allow you to view the monitoring data.

Prerequisites

Monitoring data is collected to Log Service.

Entry point

1.
2. In the **Log Application** section, click **Full-stack Monitoring**.
3. On the **Full-stack Monitoring** page, click the instance that you want to use.
4. In the left-side navigation pane, click the dashboard that you want to view below **Monitoring Kanban**.

Host monitoring

Dashboard	Description
Resource Overview	Displays the configuration data and metric data of all hosts in real time. The data includes the number of CPU cores, total disk space, average CPU utilization, and average memory usage.
Hosts	Displays the configuration data and metric data of each host in real time. The data includes the number of CPU cores, memory size, CPU utilization, and memory usage.
Hotspot Analysis	Displays the resource usage information of hotspot hosts in real time. The resources include CPUs and memory. The information includes the distribution of CPU utilization among hotspot hosts, distribution of memory usage among hotspot hosts, top CPU utilization values, and top memory usage values.
Standalone Metrics-Simple	Displays the resource usage trends of a host in real time. The resources include CPUs and memory. The trends are based on CPU utilization, disk space usage, and memory usage.
Standalone Metrics-Details	Displays the resource usage trends of a host based on the resource status in real time. The resources include CPUs and memory. A CPU can be in the following states: Total, System, User, and IOWait. Memory can be in the following states: Total, Availableused, and Used.

Kubernetes monitoring

Dashboard	Description
Resource Overview	Displays the resource usage of a Kubernetes cluster in real time. The resources include pods, hosts, Services, and Deployments.
Water-level Monitoring	Displays the resource usage information of a Kubernetes cluster in real time. The information includes the number of running pods, total number of CPUs, and file system usage.
Run-time Monitoring	Displays information about running resources of a Kubernetes cluster in real time. The information includes the number of running Deployments and number of running DaemonSets.

Dashboard	Description
Core Component Monitoring	Displays information about the core components of a Kubernetes cluster in real time. The information includes the number of objects that are stored on etcd and the queries per second (QPS) of etcd.
Nodes	Displays overall information about nodes, and the configuration data and metric data of each node in real time. The information includes the total number of nodes and total number of running pods.
Node Metrics	Displays the metric data of a node in real time. The data includes the number of pods that can be requested and CPU utilization.
Pods	Displays overall information about pods, and the configuration data and metric data of each pod in real time. The information includes the total number of pods that can be requested.
Pod Metrics	Displays the metric data of a pod in real time. The data includes the basic information about the pod and the containers in the pod.
Deployments	Displays the configuration data and metric data of each Deployment in real time. The data includes the namespace and cluster to which a Deployment belongs.
Deployment Metrics	Displays the metric data of a Deployment in real time. The data includes the proportion of the CPU limit to the total amount of CPU resources that can be requested and proportion of the memory limit to the total amount of memory resources that can be requested.
StatefulSets	Displays the configuration data and metric data of each StatefulSet in real time. The data includes the namespace and cluster to which a StatefulSet belongs.
StatefulSet Metrics	Displays the metric data of a StatefulSet in real time. The data includes the proportion of the CPU limit to the total amount of CPU resources that can be requested and proportion of the memory limit to the total amount of memory resources that can be requested.
DaemonSets	Displays the configuration data and metric data of each DaemonSet in real time. The data includes the namespace and cluster to which a DaemonSet belongs.
DaemonSet Metrics	Displays the metric data of a DaemonSet in real time. The data includes the proportion of the CPU limit to the total amount of CPU resources that can be requested and proportion of the memory limit to the total amount of memory resources that can be requested.

Middleware monitoring

Dashboard	Description
JVM Monitoring	Displays the metric data of the JVM middleware in real time. The data includes the process runtime, total memory, heap memory, and CPU utilization.

Dashboard	Description
NGINX Monitoring	Displays the metric data of the NGINX middleware in real time. The data includes the number of processed connections and QPS.
Tomcat Monitoring	Displays the metric data of the Tomcat middleware in real time. The data includes the process runtime, QPS, number of errors, and CPU utilization.
Kafka Monitoring	Displays the metric data of the Kafka middleware in real time. The data includes the status of Controller, total number of topics, and number of messages per second.
Nvidia GPU Monitoring	Displays the metric data of the NVIDIA GPU middleware in real time. The data includes the GPU utilization and memory usage.

Database monitoring

Dashboard	Description
MySQL Monitoring	Displays the metric data of MySQL databases in real time. The data includes the startup time, number of queries, and number of connections.
Redis Monitoring	Displays the metric data of Redis databases in real time. The data includes the number of cluster instances that are enabled, Redis runtime, and number of connected clients.
Elasticsearch Monitoring	Displays the metric data of Elasticsearch clusters in real time. The data includes the cluster health status and number of nodes.
ClickHouse Monitoring	Displays the metric data of ClickHouse databases in real time. The data includes Query and Merge.
MongoDB Monitoring	Displays the metric data of MongoDB databases in real time. The data includes Available Connections and Query Operations.

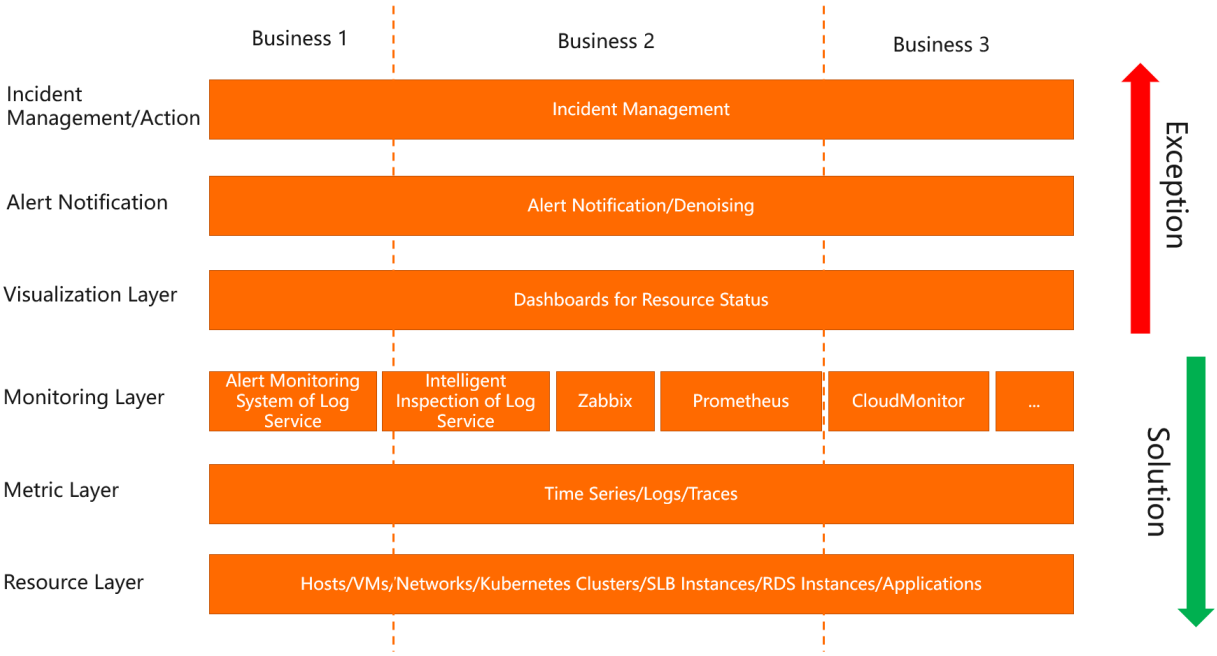
5.Alert OpsCenter

5.1. Overview of Alert OpsCenter

Alert OpsCenter is a business-centric alert management and O&M platform. You can add alerts that are generated by third-party monitoring platforms, such as Zabbix and Prometheus, and alerts that are generated by Log Service resources to a business. This way, you can use the business to manage alerts and send alert notifications in a unified manner and improve the O&M efficiency. This topic describes the architecture and features of Alert OpsCenter.

Architecture

Log Service Alert OpsCenter allows you to manage alerts by business. Each business includes a complete pipeline that starts from the resource layer and ends at incident management.



- Resource layer: includes computing, storage, and network resources, such as hosts, virtual machines, Server Load Balancer (SLB) resources, Java applications, and Go applications.
- Metric layer: includes time series data, log data, and trace data. Metrics can show the health status of each resource.
- Monitoring layer: allows you to create alert monitoring rules to monitor metrics by using monitoring tools such as Zabbix, Prometheus, the alert monitoring system of Log Service, and the intelligent inspection feature of Log Service. For example, you can monitor high CPU utilization and transient, sharp increases in network traffic.
- Visualization layer: provides visualized reports to display the alert status for different resources, such as the trends in the number of triggered alerts, the handling status of alerts, and the status of alert notifications.
- Alert notifications: If an alert is triggered, Log Service sends alert notifications based on a specified action policy. Log Service can send alert notifications to specified users by using SMS messages, voice calls, DingTalk, custom webhooks, EventBridge, and Function Compute. Before Log Service sends alert notifications, you can use alert policies to denoise alerts.
- Incident management: After alerts are sent to the alert management system, the alerts are merged

into different sets based on a route consolidation policy. An incident is automatically created for each set. O&M engineers can manage different alert incidents. For example, you can change the status of an incident to resolved, confirmed, or ignored. You can also specify incident handlers.

Features

Alert OpsCenter provides the following features:

- Alert source integration: An alert source is the source of alerts in a business. Alert sources include Log Business resources and third-party alert sources. You can use the following methods to integrate alert sources:
  - Vertical alert sources

    You can integrate alert sources based on your technical deployment. For example, if you use resources from the access layer, computing layer, and storage layer, you can add the resources to a business for unified management.
  - Horizontal alert sources

    You can integrate alert sources based on your O&M requirements. For example, if your database O&M team wants to manage all RDS instances, you can add the data of the RDS instances to a business for unified management.
  - Third-party alert sources

    If an enterprise has one or more monitoring platforms, such as Zabbix and Prometheus, the enterprise can add the alert data that is generated by the monitoring platforms to a business for unified management.
- Business policies: Alert OpsCenter allows you to configure business policies to merge, suppress, or silence alerts. Business policies support the following three configuration modes: Enable, Disable, and Mixed.
  - Enable

    In Enable mode, the alert policy and the action policy that are configured for the current business are applied. If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source are disabled.
  - Disable

    In Disable mode, the alert policy and the action policy that are configured for the current business are not applied. If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source are enabled.
  - Mixed

    If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source and the policies that are configured for the business are all enabled.
- Incident management: You can change the status of an incident to confirmed, ignored, or resolved. You can also specify incident handlers.
- Alert Status dashboard: Alert OpsCenter provides the **Alert Status** dashboard that displays the status of an alert source, or the details of triggered alerts and alert status in a business.
- Troubleshooting dashboards: Alert OpsCenter provides troubleshooting dashboards that include the

following dashboards: Global Alert Pipeline Center, Global Alert Rule Center, Global Alert Troubleshooting Center, and Pub Alert Center. The preceding dashboards display information about alerts.

5.2. Integrate alerts

Log Service Alert OpsCenter allows you to manage alerts by business. You can add one or more alert sources that you want to monitor to a business for unified management. This topic describes how to integrate alerts. To integrate alerts, you must add a business, add alert sources, configure an alert monitoring rule, and then configure a business policy.

Step 1: Add a business

1.
2. On the **Intelligent O&M** tab of the **Log Application** section, click **Alert OpsCenter**.
3. On the **Alert OpsCenter** page, click **Add New Business**.
4. In the **Add Business** dialog box, set the **Business ID** and **Business Name** parameters and click **OK**.

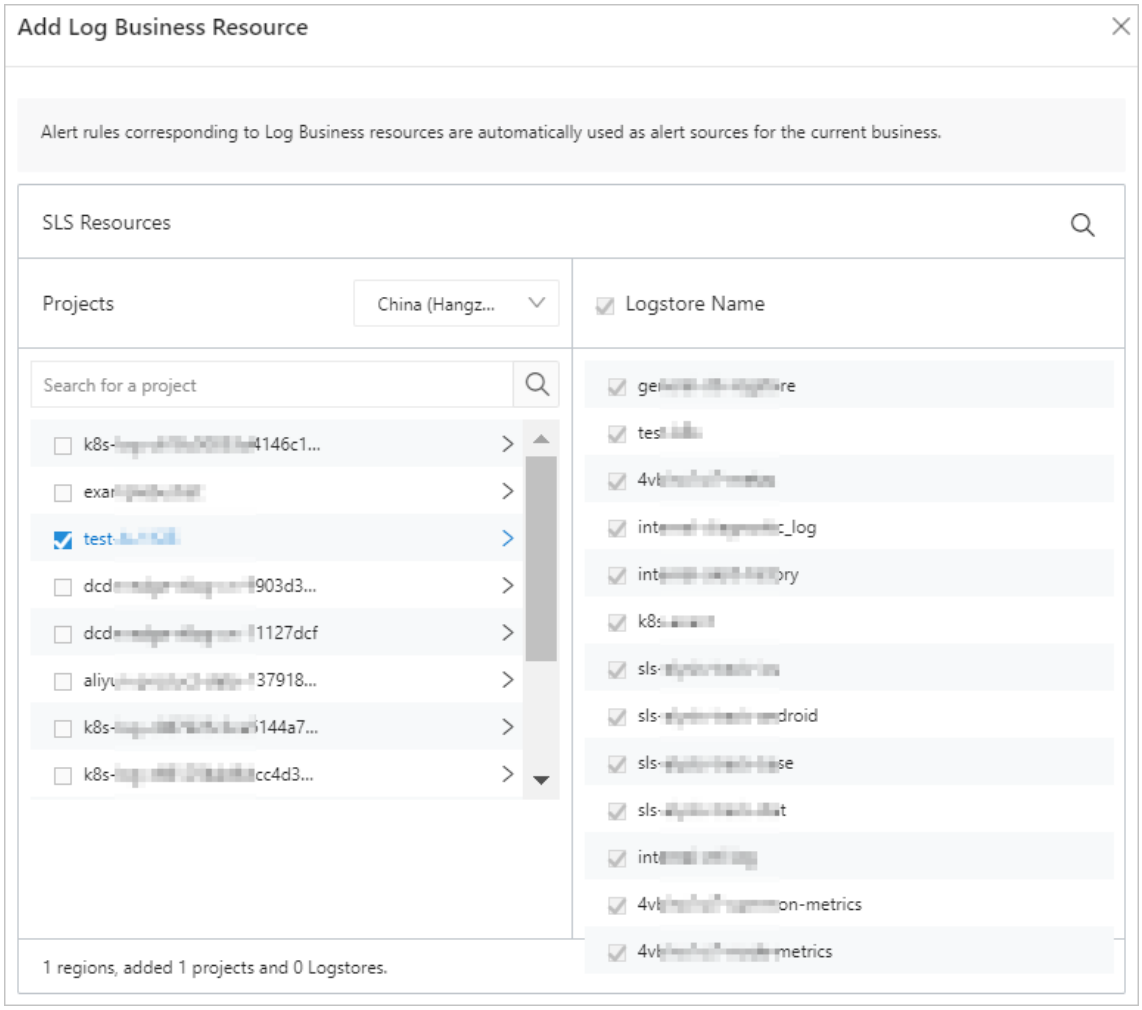
Step 2: Add alert sources

1. On the **Alert OpsCenter** page, find the business that you want to manage and click the ID of the business.
2. In the left-side navigation pane, choose **Alert Management > Integration**.
3. On the **Integration** page, click Add in the card of an alert source that you want to integrate.
4. Configure an alert source.

   Alert OpsCenter supports the following types of alert sources:

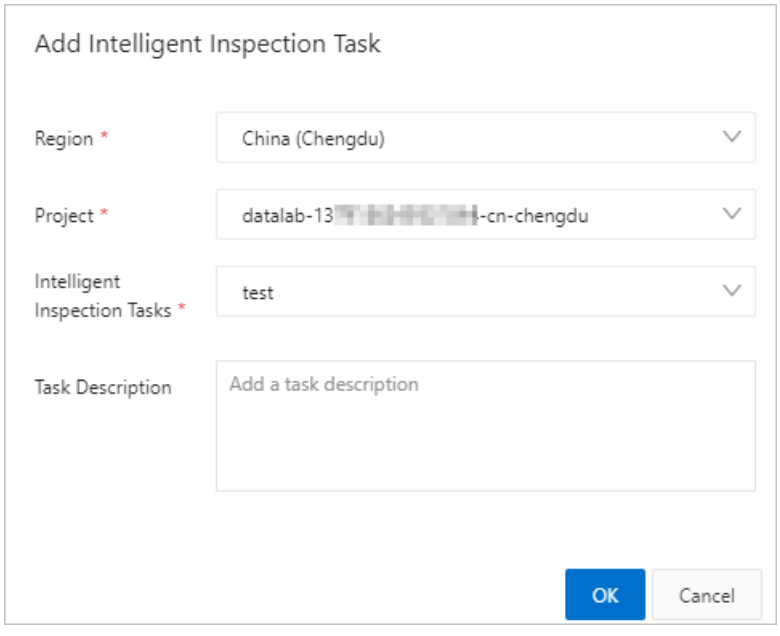
   - Log Business resources

     In the **Add Log Business Resource** panel, select the projects that you want to add and click **OK**.

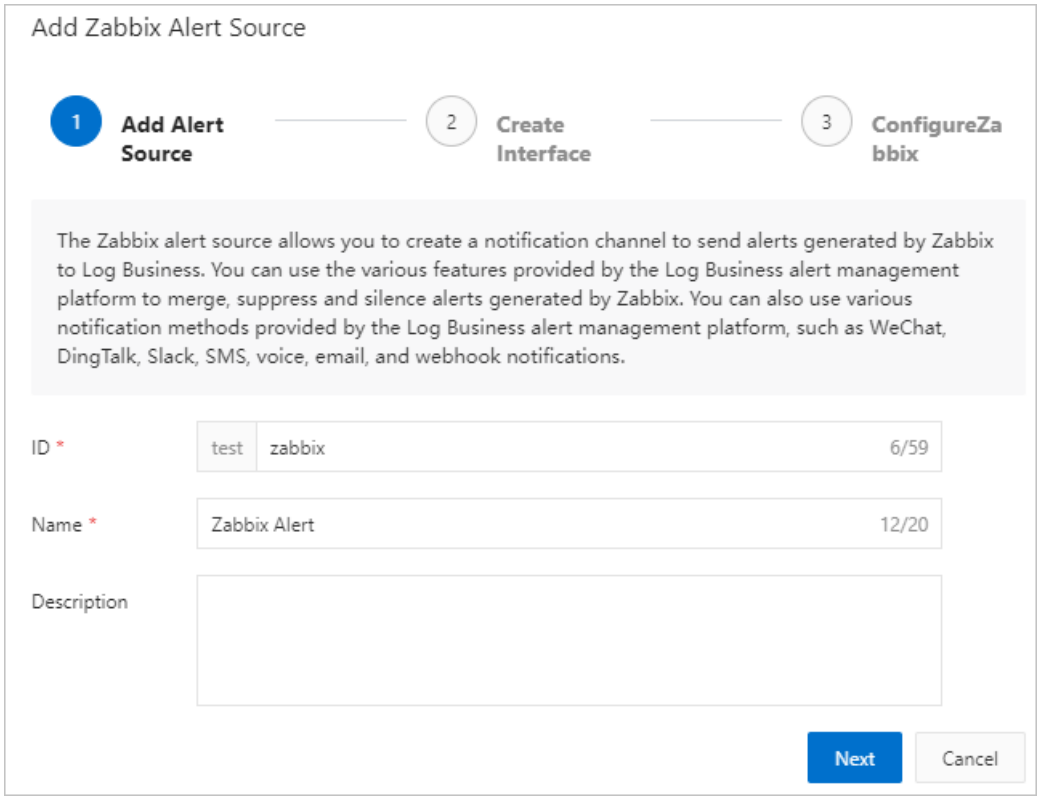


- Intelligent inspection

  In the **Add Intelligent Inspection Task** dialog box, select a region, a project, and an intelligent inspection task. Then, click **OK**.



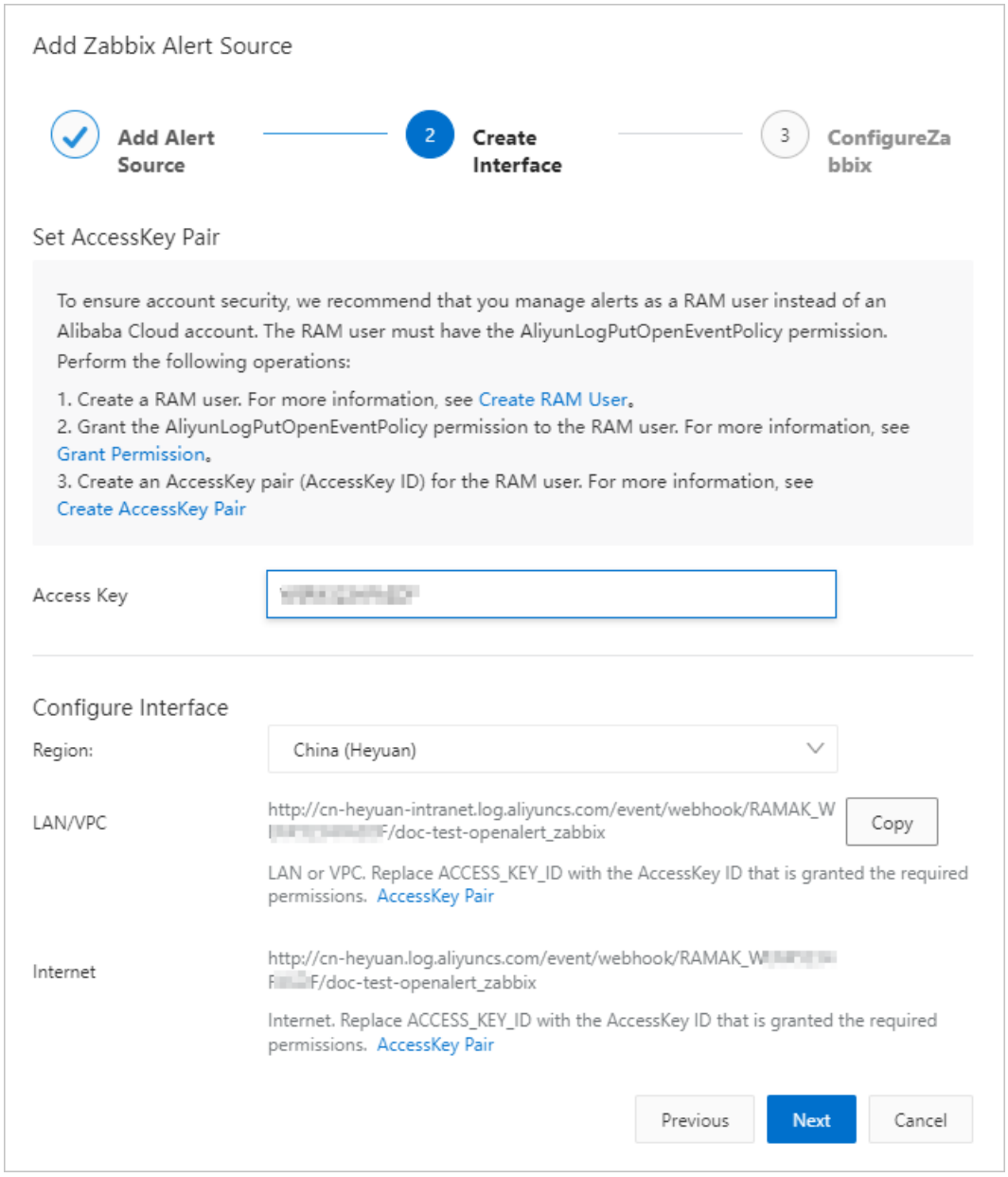
- Third-party alert sources
  a. In the **Add Alert Source** step, specify the ID and name of the alert source. Then, click **Next**.



  b. In the **Create Interface** step, set the **Access Key** and **Region** parameters.

     Set the **Access Key** parameter to the AccessKey ID of an Alibaba Cloud account or a RAM user based on your business requirements. For information about how to obtain an AccessKey ID, see Appendix: Obtain an AccessKey ID.

     Then, move the pointer over the blank area next to the text box and copy the full URL, domain name, or subpath of the webhook URL. The URL is required when you configure parameters on a third-party alerting platform.



c. Complete the required settings on the corresponding third-party alerting platform.

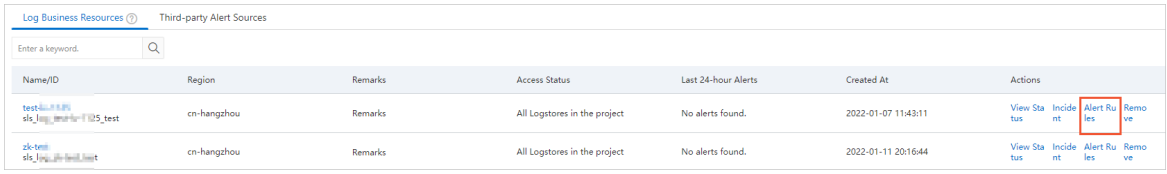
For more information, see the following topics:

- Configure Datadog
- Configure Zabbix
- Configure New Relic
- Configure the alert contact and an alert notification method in the CloudMonitor console
- Configure Loki
- Configure Grafana
- Configure Prometheus
- Configure Alertmanager
- Configure CloudWatch

Step 3: Configure an alert monitoring rule

If you add Log Business resources, you must configure an alert monitoring rule for the resources.

1. On the **Log Business Resources** tab of the **Integration** page, find the alert source and click **Alert Rules** in the Actions column.



2. On the **Alert Rules/Incidents** tab, configure an alert monitoring rule.

   You can use a built-in alert monitoring rule of Log Service. Find the alert monitoring rule and click **Enable**.

   If built-in alert monitoring rules do not meet your business requirements, you can click **Create Alert** to create a custom alert monitoring rule. For more information, see Create an alert monitoring rule for logs.

Step 4: Configure a business policy

1. In the left-side navigation pane, choose **Alert Management > Service Policy**.
2. In the **Business Policy** section, select a configuration mode for the business policy.
   i. Click **Modify Settings**.

   You can select **Enable**, **Disable**, or **Mixed** based on your business requirements. The following table describes the preceding modes.

Mode	Description
Enable	In Enable mode, the alert policy and the action policy that are configured for the current business are applied. If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source are disabled.
Disable	In Disable mode, the alert policy and the action policy that are configured for the current business are not applied. If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source are enabled.
Mixed	If an alert source in the business is associated with an available alert policy and an available action policy in Alert Center, the policies that are associated with the alert source and the policies that are configured for the business are all enabled.

   ii. Click **Save**.
3. In the **Notification Management** section, configure an alert policy and action policy.
   - Simple mode

If you select **Simple Mode**, Log Service uses the built-in alert policy sls.builtin.dynamic to manage alerts. You need only to configure an action group. For more information, see Notification methods.

After you configure the action group, Log Service automatically creates an action policy named *Rule name*-`Action policy`.

- Advanced mode

  If you select **Advanced Mode**, you can configure an alert policy and an action policy. An alert policy includes the following child policies: route consolidation, suppression, and silence. An action policy includes the following settings: notification method, alert escalation, and notification method quota. For more information, see Create an alert policy, Create an action policy, and Configure notification quotas.

5.3. Manage alert incidents

After alerts are sent to the alert management system, the alerts are merged into different sets based on a route consolidation policy. An incident is automatically created for each set. Alert incidents help you manage each alerting process and workflow. This topic describes how to manage alert incidents.

Prerequisites

Alerts are integrated. For more information, see Integrate alerts.

Procedure

1.
2. On the **Intelligent O&M** tab of the **Log Application** section, click **Alert OpsCenter**.
3. On the **Alert OpsCenter** page, find the business that you want to manage in the business list and click the ID of the business.
4. In the left-side navigation pane, choose **Alert Status > Incidents**.
5. In the Incidents list, find the incident that you want to manage. You can click OK, Set Handler, or Ignore in the Actions column to manage the incident.

   For example, if an incident is in the **Pending Evaluation** state, you can click **OK** and specify a handler to handle the incident.

   For more information about alert incidents, see Alert incident management.

5.4. View the Alert Status dashboard

The Alert Status dashboard displays the status of an alert source, or the details of triggered alerts and the alert status in a business.

Prerequisites

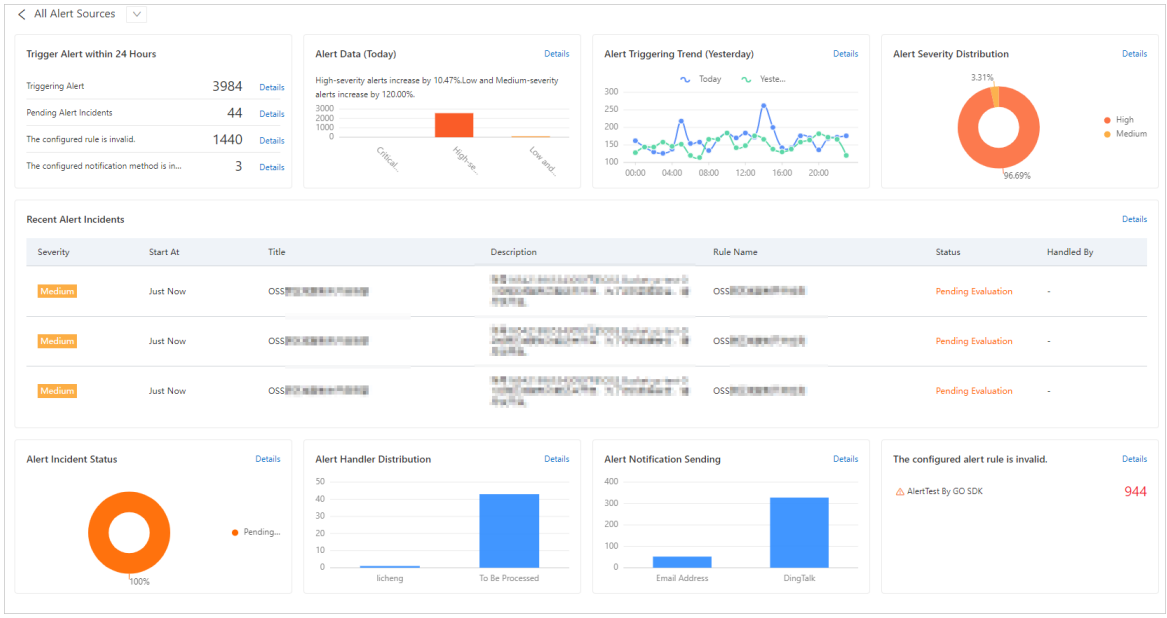
Alerts are integrated. For more information, see Integrate alerts.

Go to the dashboard page

1.
2. On the **Intelligent O&M** tab of the **Log Application** section, click **Alert OpsCenter**.
3. On the **Alert OpsCenter** page, find the business that you want to manage in the business list and

click the ID of the business.

4. In the left-side navigation pane, click **Alert Status**.



Data details

On the **Alert Status** page, you can view the details of triggered alerts and the alert status within a specified period of time.

Chart	Description
Trigger Alert within 24 Hours	Displays the number of alerts that are triggered within 24 hours in the specified alert source of the current business.
Alert Data (Today)	Displays the number of alerts and the distribution of alert severities for the current day in the specified alert source of the current business.
Alert Triggering Trend (Yesterday).	Displays the result of comparison between the amount of alert data of the current day and the amount of alert data of the previous day in the specified alert source of the current business.
Alert Severity Distribution	Displays the distribution of alert severities in the specified alert source of the current business.
Recent Alert Incidents	Displays the details of alert incidents in the specified alert source of the current business.
Alert Incident Status	Displays the distribution of incident status in the specified alert source of the current business.
Alert Handler Distribution	Displays the distribution of incident handlers in the specified alert source of the current business.
Alert Notification Sending	Displays the distribution of alert notification methods in the specified alert source of the current business.

Chart	Description
The configured alert rule is invalid	Displays the configuration errors of related alert monitoring rules in the specified alert source of the current business.

5.5. View troubleshooting dashboards

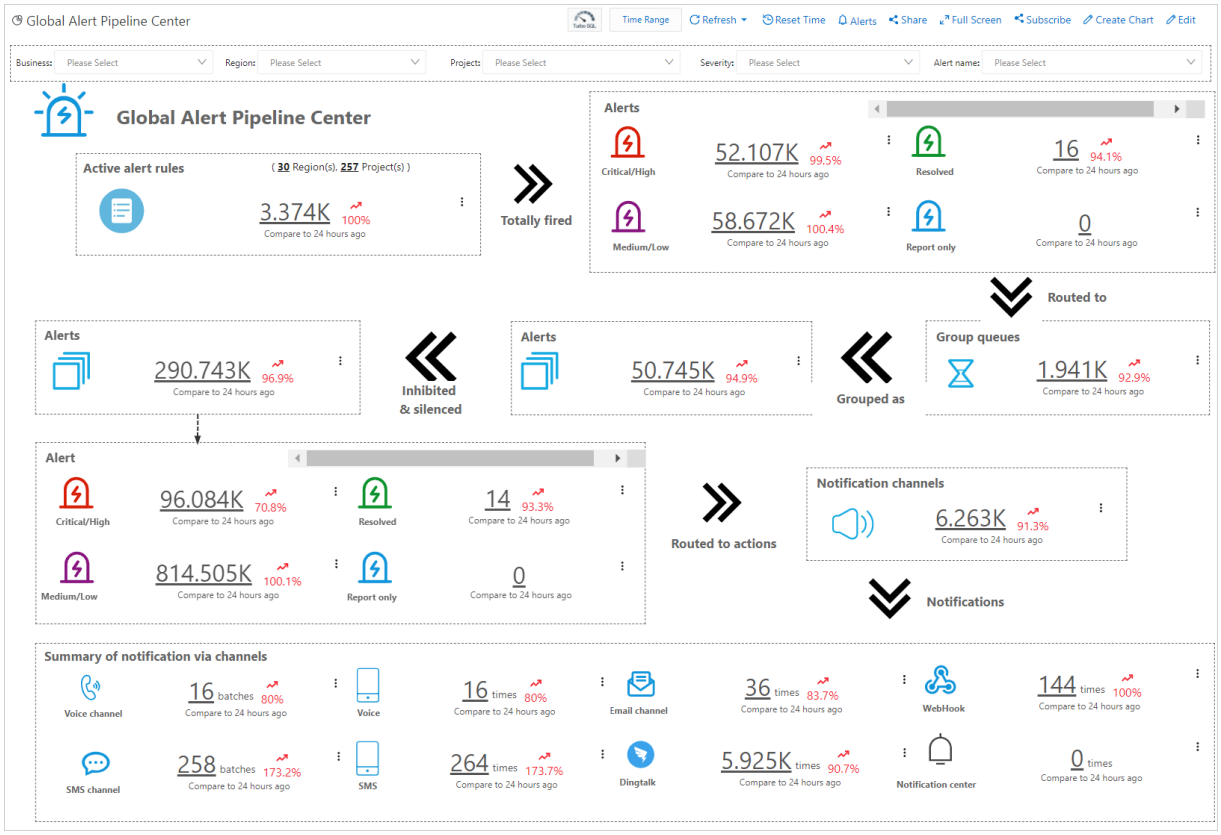
Troubleshooting dashboards include the following dashboards: Global Alert Pipeline Center, Global Alert Rule Center, Global Alert Troubleshooting Center, and Pub Alert Center. The preceding dashboards display information about alerts.

Go to the dashboard page

1.
2. On the **Intelligent O&M** tab of the **Log Application** section, click **Alert OpsCenter**.
3. On the **Alert OpsCenter** page, find the business that you want to manage in the business list and click the ID of the business.
4. In the left-side navigation pane, choose **Alert Status > Troubleshooting**.

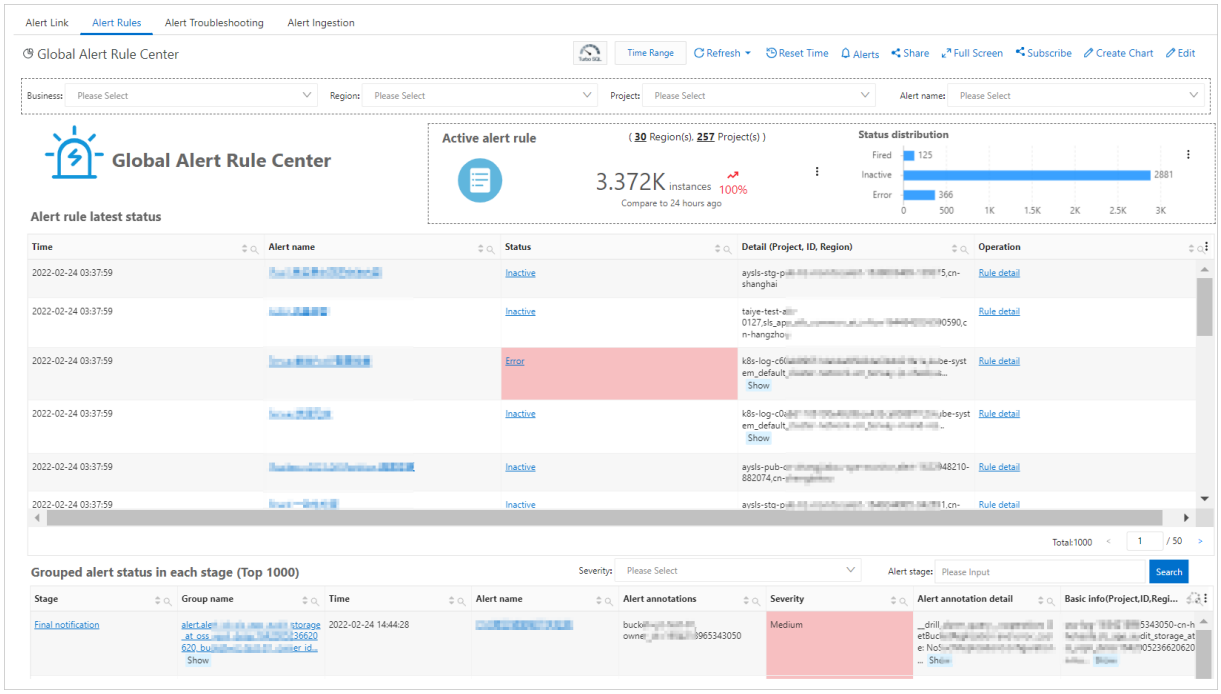
Alert link

The **Global Alert Pipeline Center** dashboard displays the historical information about the alerts that are triggered within a specified period of time for the current Alibaba Cloud account. The historical information includes how alerts are triggered, how alerts are denoised, and how alert notification are sent. The dashboard also displays all historical data. The data includes the number of enabled alert monitoring rules, the number of alerts by severity, the number of merged alerts, the number of deduplicated alerts, the number of silenced alerts, the number of alert notifications that are sent, and the number of alert notifications for each notification method.



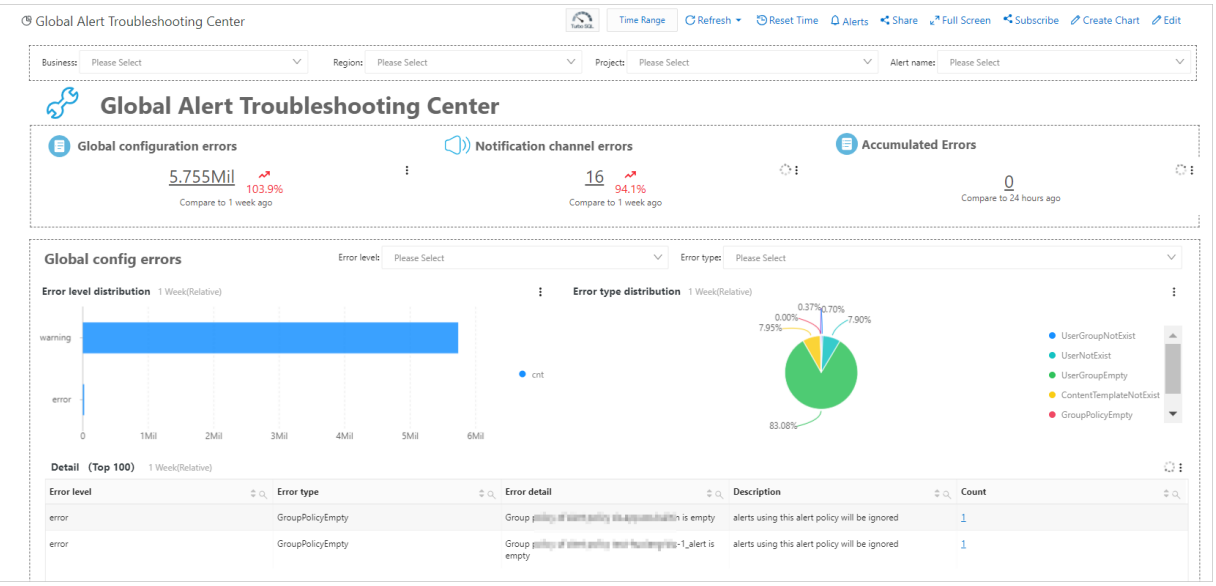
Alert monitoring rules

The **Global Alert Rule Center** dashboard displays all alert monitoring rules within a specified period of time for the current Alibaba Cloud account. The dashboard also displays the number of enabled rules, the distribution of rule status, and the details of rules.



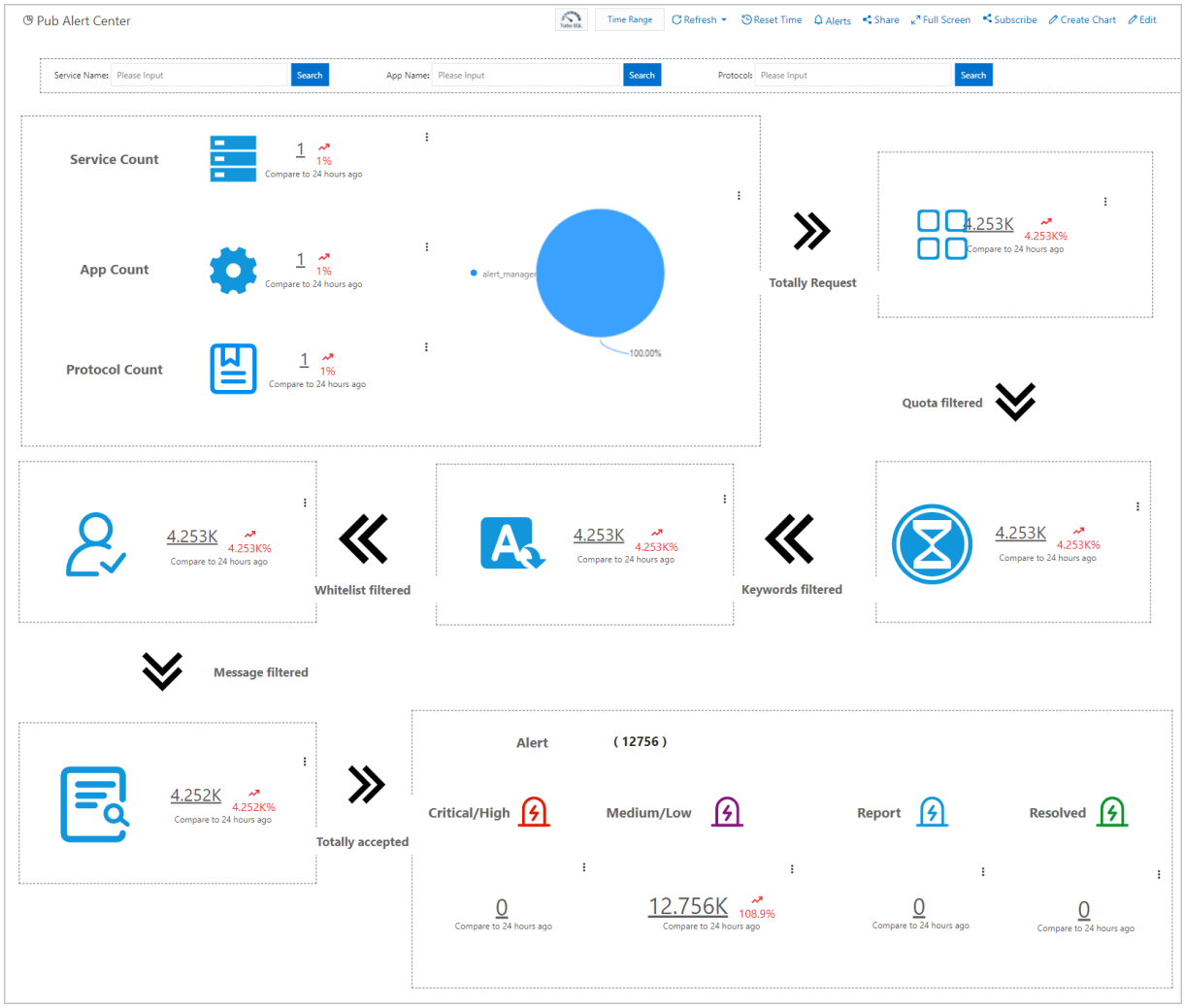
Alert troubleshooting

The **Global Alert Troubleshooting Center** dashboard displays information about alert configuration errors that occur within a specified period of time for the current Alibaba Cloud account. The information includes the number and details of global configuration errors, the number and details of configuration errors for each notification method, and the number and details of configuration errors for alert monitoring rules.



Alert ingestion

The **Pub Alert Center** dashboard displays the data of the alert ingestion system within a specified period of time for the current Alibaba Cloud account. The data includes the number of active alert ingestion services, the number of alert ingestion applications, the number of protocols, the number of requests that are sent by active alert ingestion services, and the number of ingested alerts.


Pub Alert Center
Time Range
Refresh
Reset Time
Alerts
Share
Full Screen
Subscribe
Create Chart
Edit
Service Name: Please Input
Search
App Name: Please Input
Search
Protocol: Please Input
Search
Service Count
1
1%
Compare to 24 hours ago
App Count
1
1%
Compare to 24 hours ago
Protocol Count
1
1%
Compare to 24 hours ago
alert_manager
100.00%
Totally Request
4.253K
4.253K%
Compare to 24 hours ago
Quota filtered
4.253K
4.253K%
Compare to 24 hours ago
Whitelist filtered
4.253K
4.253K%
Compare to 24 hours ago
Keywords filtered
4.253K
4.253K%
Compare to 24 hours ago
Message filtered
4.252K
4.252K%
Compare to 24 hours ago
Totally accepted
Alert
(12756)
Critical/High
Medium/Low
Report
Resolved
0
Compare to 24 hours ago
12.756K
108.9%
Compare to 24 hours ago
0
Compare to 24 hours ago
0
Compare to 24 hours ago

6.K8s Event Center

6.1. Create and use an event center

K8s Event Center records the status changes of Kubernetes clusters. For example, K8s Event Center records a status change when you create, run, or delete a pod, or when a component exception occurs. K8s Event Center aggregates all the events in Kubernetes clusters in real time. This allows you to perform various operations on the events. For example, you can store, query, analyze, and visualize event data and configure alerts for the events. This topic describes how to create and use an event center in K8s Event Center.

Billing

If the following conditions are met, you are not charged when you use an event center:

- The data retention period of the Logstore that is associated with your event center is 90 days, which is the default value.
- The amount of data that is written to your event center per day is less than 256 MB, which is equivalent to approximately 250,000 events.

Examples:

- If the default data retention period is specified for the associated Logstore and 1,000 events are generated in your Kubernetes cluster per day, you can use your event center for free.
- If you change the data retention period to 105 days and 1,000 events are generated in your Kubernetes cluster per day, you are charged only for the data that is stored in the associated Logstore for a period that exceeds 90 days. The fee is calculated based on the billable item of the storage space occupied by log data. For more information about the billable item, see Billable items.

Step 1: Create an event center

1. Log on to the Log Service console.
2. In the **Log Application** section, click the **Intelligent O&M** tab and click **K8s Event Center**.
3. On the **Event Center Management** tab, click **Add**.
4. In the **Add Event Center** panel, configure the parameters.
   - If you select **Existing Project**, you can select an existing project from the **Project** drop-down list to manage the resources of your event center. The resources include a Logstore and the related dashboards.
   - If you select **Kubernetes Cluster**, you can select an existing Kubernetes cluster from the **Kubernetes Cluster** drop-down list. If you use this method to create an event center, Log Service automatically creates a project whose name is in the k8s-log-{cluster-id} format to manage the resources of your event center. The resources include a Logstore and the related dashboards.

   > **Note** After you create an event center, Log Service automatically creates a Logstore named k8s-event in the specified project. Associated dashboards are also created for the event center.

5. Click **Next**.

Step 2: Deploy the eventer and node-problem-detector components

Before you can use an event center, you must deploy the eventer and node-problem-detector components in your Kubernetes cluster.

- Deploy the eventer and node-problem-detector components in a Container Service for Kubernetes (ACK) cluster

  The ack-node-problem-detector component that is provided in the Marketplace of ACK is integrated with the features of the eventer and node-problem-detector components. Therefore, to deploy the eventer and node-problem-detector components in an ACK cluster, you need only to deploy the ack-node-problem-detector component. For more information about how to deploy the ack-node-problem-detector component, see Event monitoring.

  i. Log on to the ACK console.
  ii. In the left-side navigation pane, choose **Marketplace > Marketplace**.
  iii. On the **App Catalog** tab, search for and click **ack-node-problem-detector**.
  iv. Click **Deploy**.
  v. In the **Basic Information** step, select your cluster and click **Next**.
  vi. In the **Parameters** step, set the parameters of the eventer node and click **OK**.
    - enabled: Set the value of the enabled parameter under **eventer > sinks > sls** to true.
    - topic: Optional. Set the value to your cluster name. The name can contain lowercase letters, underscores (_), and hyphens (-).
    - project: Set the value to the name of the project that is specified when you create the event center.
    - logstore: Set the value to k8s-event.

```
  sinks:
     sls:
       enabled: true
       # If you want the monitoring results to be notified by sls, set enabled to tru
e.
       topic: "my-cluster"
       project:  "{sls-project-name}"
       # You can view the project information by logging in to the
       # SLS console. Please fill in the name of the project here.
       # eg: your project name is k8s-log-cc18a5f3443dhdss22654da,
       # then you can fill k8s-log-cc18a5f3443dhdss22654da to project label.
       logstore: "k8s-event"
       # You can view the project information by logging in to the
       # SLS console. Please fill the logstore address in here.
```

- Deploy the eventer and node-problem-detector components in a self-managed Kubernetes cluster
  i. Deploy the eventer component. For more information, see Collect Kubernetes events.
  ii. Deploy the node-problem-detector component. For more information, visit GitHub.

Step 3: Use an event center

After you create an event center in K8s Event Center and deploy the eventer and node-problem-detector components, you can use the event center. For example, you can use the event center to view event statistics, query event details, view the lifecycle of a pod, view node events, view core component events, configure alerts, perform custom queries, and update the version of K8s Event Center.

In the left-side navigation pane of the **K8s Event Center** page, find the event center that you want to manage and click the > icon. Then, you can perform the following operations.

Operation	Description
View event statistics	The **Event Overview** tab displays the statistics of core events. The statistics include the total number of events, the difference between the number of error events within the current day and the preceding day, statistics of alert items, trends of error events, and details of pod OOM events. **Note** If a pod OOM event is recorded, you can view only the node on which the event occurs, process name, and process ID. The pod in which the event occurs cannot be located. However, you can query the pod restart event that occurs around the time of the pod OOM event. This way, you can locate the pod.
Query event details	The **Event Details** tab displays the details about the events that are returned by using different filter conditions, such as the event type, event destination, host, namespace, and name.
View the lifecycle of a pod	The **Pod Lifecycle** tab displays the details about the events that occur within the lifecycle of a pod. You can filter important pod events by event level.
View node events	The **Node Event** tab displays the details about node events. You can view the lifecycle of a node and the events that occur on the node.
View core component events	The **Event Core** tab displays the details about core component events. The events include NLC.Task.RestartECS.Fail and NLC.Task.URL.Mode.Unimplemented.
Configure alerts	On the page that appears after you click **Alert Configuration**, you can configure alerts for your event center. For more information, see Configure alerts.

Operation	Description
Perform custom queries	On the page that appears after you click **Custom Query**, you can execute custom query statements. All events in an event center are stored in a Logstore. You can use all features of the Logstore. For example, you can perform custom queries, consume event data, create custom reports, or configure custom alerts. For more information, see Query and analyze logs. If you want to access a project that is specified for an event center, you can obtain the name of the project by using one of the following methods: • Obtain the name of the project by using the URL of the page that appears after you click **Custom Query**. The URL is in the `https://sls.console.aliyun.com/lognext/app/k8s-event/project/k8s-log-xxxx/logsearch/k8s-event` format. The field that follows the Project field indicates the name of the project. Example: k8s-log-xxxx. • In the list of event centers on the **Event Center Management** tab, find the event center and view the project name.
Update the version of K8s Event Center	On the page that appears after you click **Version Update**, you can update the version of K8s Event Center.

Delete an event center

On the **K8s Event Center > Event Center Management** page, find the event center that you want to delete and click the icon in the Actions column.

FAQ

- Why does no data exist in my event center?

  After you deploy an event center, new events are automatically collected to the event center. You can click **Custom Query** to search for the events. We recommend that you set the time range in the upper-right corner to 1 Day. Data may not be found in the event center due to the following reasons:

  - After you deploy the event center, no events are generated in the associated Kubernetes cluster.

    You can run the `kubectl get events --all-namespaces` command to check whether new events are generated in a cluster.

- Invalid values are specified for the parameters that are used to deploy the eventer and node-problem-detector components.
  - If you are using an ACK cluster, perform the following steps:
    a. Log on to the ACK console.
    b. On the **Clusters** page, find the cluster that you want to manage and click the name of the cluster.
    c. In the left-side navigation pane, choose **Applications > Helm**.
    d. On the **Helm** page, find **ack-node-problem-detector** and click **Update** in the Actions column.
    e. Check and modify the parameter settings. For more information, see Step 2: Deploy the eventer and node-problem-detector components.
  - If you are using a self-managed Kubernetes cluster, check the configurations based on the descriptions in Collect Kubernetes events.

- How do I view the logs of a container in which an event occurs?
  - If you are using an ACK cluster, perform the following steps:
    a. Log on to the ACK console.
    b. On the **Clusters** page, find the cluster that you want to manage and click the name of the cluster.
    c. In the left-side navigation pane, choose **Workloads > Pods**.
    d. Set the **Namespace** parameter to **kube-system**.
    e. On the **Pods** page, find the pod that you want to manage and click **Logs** in the Actions column.
  - If you are using a self-managed Kubernetes cluster, filter the **kube-system** namespace and find the file whose name is prefixed with eventer-sls to view the pod logs.

6.2. Configure alerts

Log Service provides built-in alert monitoring rules. If you want to monitor a Kubernetes cluster in real time, you need only to add an alert instance. An alert instance allows Log Service to send alert notifications by using different methods such as DingTalk. This topic describes how to configure alerts.

Prerequisites

An event center is created in K8s Event Center, and Kubernetes cluster events are collected to the event center. For more information, see Create and use an event center.

Context

K8s Event Center provides the following built-in resources: alert monitoring rules, ACK action policy, ACK user group, ACK-pod alert template, ACK alert template, ACK-node alert template, and ACK-object alert template. The built-in resources can meet the requirements of most alerting scenarios. Before you use the built-in resources, take note of the following items:

- You can specify the ACK alert policy in an alert monitoring rule.
- You can specify the ACK user group and an alert template in the ACK alert policy. The alert template can be the ACK-pod alert template, ACK alert template, ACK-node alert template, or ACK-object alert template.

After an alert is triggered, Log Service sends an alert notification to the specified users based on the action policy.

Step 1: Create a user

1.
2. In the **Log Application** section, click the **Intelligent O&M** tab and click **K8s Event Center**.
3. In the left-side navigation pane, click the icon of the event center that you want to manage.

   Then, click **Alert Configuration**.
4. On the **Alert Center** page, choose **Alert Management > User Management**.
5. Create a user.

   For more information, see Create users.

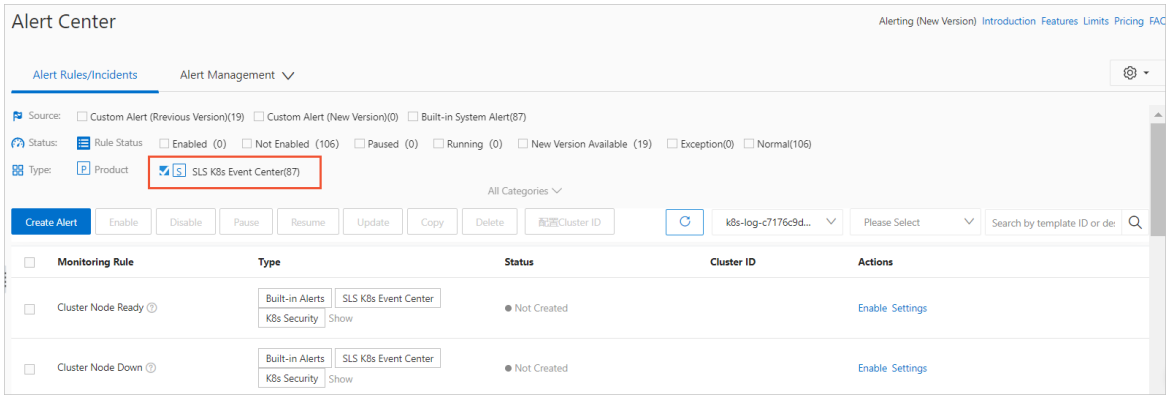
Step 2: Add the user to the ACK user group

1. On the **Alert Center** page, choose **Alert Management > User Group Management**.
2. In the user group list, click **Edit** for **sls.app.ack.builtin**.
3. In the **Edit User Group** dialog box, add the user that you create from the **Available Members** section to the **Selected Members** section. Then, click **OK**.

Step 3: Add an alert instance

Log Service provides dozens of built-in alert monitoring rules for K8s Event Center. You need only to add an alert instance based on your business requirements. In the following example, an alert instance is added to the alert monitoring rule **Cluster Node Ready**.

1. On the **Alert Rules/Incidents** tab of the **Alert Center** page, click **SLS K8s Event Center**.



2. In the alert monitoring rule list, find Cluster Node Ready and click **Settings** in the Actions column.
3. In the **Parameter Settings** dialog box, configure the following parameters and click **Save and Enable**.

Parameter	Description
ACK Cluster ID	Enter the ID or name of a Kubernetes cluster. The cluster is the one that you use when you deploy the eventer and node-problem-detector components. For more information, see Deploy eventer and node-problem-detector components.

Parameter	Description
Action Policy	Select an action policy for the alert monitoring rule. Log Service sends alert notifications to the specified users based on this action policy. Default value: `sls.app.ack.builtin`, which indicates the ACK action policy. You can also create a custom action policy. For more information, see Create an action policy.
Repeat Interval	Specify a period to prevent repeated notifications. In this period, Log Service does not notify you of repeated alerts. Examples: 1d, 2h, and 3m, which indicate 1 day, 2 hours, and 3 minutes.
Severity	Specify the severity of the alert message.

4. Click **Save and Enable**.

What to do next

After you configure the alerts for an event center, you can perform the following operations.

Operation	Description
Disable an alert instance	If you disable an alert instance, the value in the **Status** column of the alert instance changes to **Not Enabled**, and alerts are no longer triggered based on the alert instance. The configurations of the alert monitoring rule are not deleted. If you want to enable the alert instance again, you do not need to reconfigure the parameters of the alert monitoring rule.
Pause an alert instance	If you pause an alert instance, alerts are not triggered based on the alert instance within a specified period of time.
Delete an alert instance	If you delete an alert instance, the value in the **Status** column of the alert instance changes to **Not Created**. The configurations of the alert monitoring rule are deleted. If you want to enable the alert instance again, you must reconfigure the parameters of the alert monitoring rule.
Reconfigure an alert instance	You can reconfigure the parameters of an alert instance.
View	You can view the general information and historical report of an alert.
Follow	You can add an alert instance to the list that you follow.
Customize an alert monitoring rule	If a built-in alert monitoring rule does not meet your business requirements, you can click **Create Alert** to create a custom alert monitoring rule. For more information, see Create an alert monitoring rule for logs.

7.CloudLens for SLS

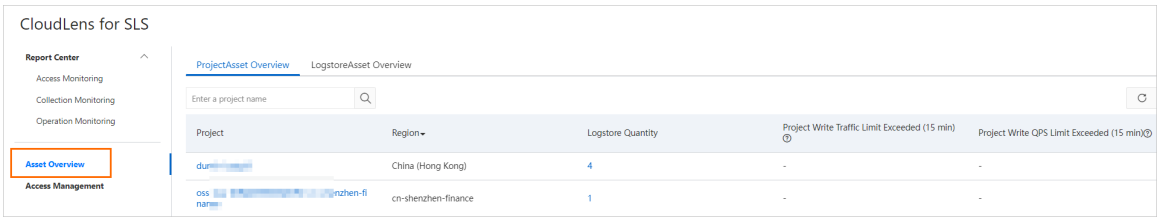
7.1. Usage notes

Log Service provides the application to help you monitor and manage assets such as projects and Logstores. This way, you can manage the assets in a more efficient manner. For example, you can view data provided by the application to check the consumption of assets.

Features

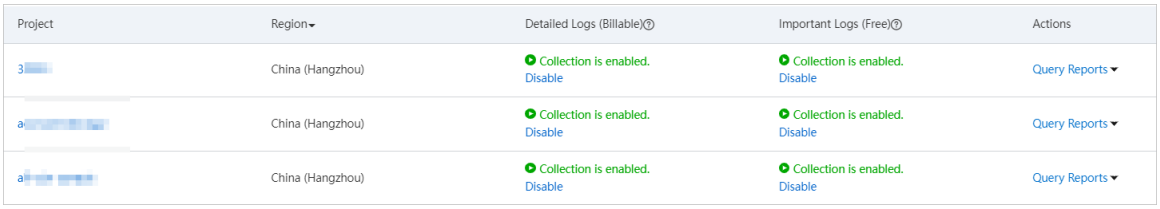
supports the following features:

- The application allows you to manage all projects and Logstores that match specified conditions within your Alibaba Cloud account in a centralized manner. For more information about conditions, see Limits.

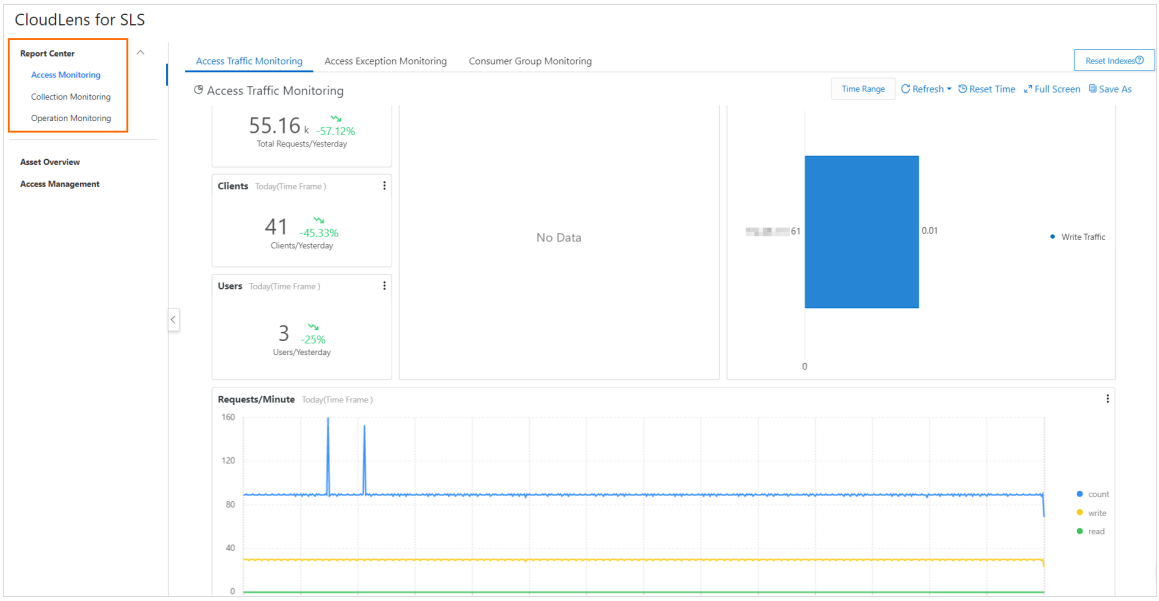


- The application allows you to enable log collection for important logs and detailed logs with a few clicks and manage the log collection status in a centralized manner.
  - Important logs record the consumption delay of consumer groups for each Logstore, and record the errors, heartbeats, and statistics of Logtail. The logs are stored in a Logstore named internal-diagnostic_log.
  - Detailed logs record the operations to create, modify, update, and delete resources in each project. Detailed logs also record data read and write operations. The logs are stored in a Logstore named internal-operation_log.

  For more information, see Service logs.



- The application provides the following types of reports: Access Monitoring, Collection Monitoring, and Operation Monitoring. For more information, see View data reports.



Assets

- Projects and Logstores
  - When you enable log collection for important logs, you can select a project based on your business requirements. After log collection is enabled for important logs, Log Service generates a Logstore named internal-diagnostic_log in the project.
  - When you enable log collection for detailed logs, you can select a project based on your business requirements. After log collection is enabled for detailed logs, Log Service generates a Logstore named internal-operation_log in the project.
- Dashboards

Dashboard		Description
Access Monitoring	Access Traffic Monitoring	Displays information about the access to Log Service in charts. The charts include Total Requests, Clients, Users, Top 10 IPs with Most Read Traffic, Top 10 IPs with Most Write Traffic, Requests, and Write Traffic.
	Access Exception Monitoring	Displays information about abnormal access to Log Service in charts. The charts include Total Requests, Percentage of Failed Requests, Quota Exceeded, Error Status Distribution, Logstore with Excessive Write Traffic, Error Requests, and Trend of Request Latency.
	Consumer Group Monitoring	Displays information about consumer groups in charts. The charts include Consumer Group, Logstore, Shard, Fall Behind, Data, Consumer Group List, Top 10 Latency, and Trend of Consumer Group Latency.
	Logtail Overall Status	Displays information about Logtail in charts. The charts include Active Logtail, Data Traffic, Status, CPU, Logtail Overall Status, Trend of CPU, Trend of Memory, and Data Sending Traffic.

Dashboard		Description
Collection Monitoring	Logtail File Collection Monitoring	Displays information about the files from which you want to collect logs in charts. The charts include Logtail File Collection, Machine Collection, Collection File Distribution, Trend of Log Collection, Trend of Average Latency, Trend of Parse Failure, and Trend of Send.
	Logtail Exception Monitoring	Displays information about Logtail exceptions in charts. The charts include Active Logtail, Restart Logtail List, Restart Clients, and Critical Error.
Operation Monitoring	Operation Monitoring	Displays information about operation records and the quotas that are exceeded in charts. The charts include Logstore Related Operation Count, Logstore Related Fail Operation Count, Top 10 Operation, Top 10 Fail Operation Error Code, LogStore Quota Exceeded, Dashboard Quota Exceeded, Shard Quota Exceeded, Alert Quota Exceeded, Machine Group Quota Exceeded, and Logtail Config Quota Exceeded.

Billing

- You are not charged when you collect, store, query, and analyze important logs. You are charged for data shipping and data transformation based on the pay-as-you-go billing method.
- The billing method for all Logstores is the same regardless of whether a Logstore contains detailed logs.

For more information, see Billable items.

Limits

CloudLens for SLS is supported in the following regions: China (Qingdao), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Chengdu), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Hong Kong), Russia (Moscow), India (Mumbai), UK (London), Australia (Sydney), Germany (Frankfurt), UAE (Dubai), US (Silicon Valley), US (Virginia), Indonesia (Jakarta), Malaysia (Kuala Lumpur), Philippines (Manila), Singapore (Singapore), and Japan (Tokyo).

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

Notice If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

7.2. Grant the operation permissions on to a RAM user

This topic describes how to grant the operation permissions on to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant the operation permissions on to a RAM user in one of the following modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode allows you to perform fine-grained access control. However, this mode requires complex configurations.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the code editor with one of the following scripts, and then click **Next: Edit Basic Information**.

   You can grant the read-only permissions or read and write permissions on to the RAM user.

- Read-only permissions: Use the following script to authorize the RAM user only to view each page of .

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "log:GetProductDataCollection",
            "Resource": [
                "acs:log:*:*:project/*/logstore/*"
            ],
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

- Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by .

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
```

```
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:GetProductDataCollection",
                "log:OpenProductDataCollection",
                "log:CloseProductDataCollection"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.sls.global_conf.sta
ndard_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
                    "ram:ServiceName": "audit.log.aliyuncs.com"              }
            }
        }
    ],
    "Version": "1"
}
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the policy name to log-sls-policy.

3. Attach the policy to the RAM user.

i. In the left-side navigation pane, choose **Identities > Users**.

ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.

iii. In the **Select Policy** section of the **Add Permissions** panel, click **Custom Policy**. Then, click the policy that you create in Step . In this example, click log-sls-policy.

iv. Click **OK**.

7.3. Enable the log collection feature

allows you to enable the log collection feature for the important logs and detailed logs of Log Service with a few clicks. This topic describes how to enable the log collection feature. This topic also describes the operations that you can perform after you enable the feature.

Prerequisites

A project is created. For more information, see Create a project.

Authorization

Notice You need to perform this operation only once.

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click .
3. Follow the on-screen instructions to enable .

   When you enable the application, Log Service automatically authorizes to assume the AliyunServiceRolePolicyForSLSAudit service-linked role to collect important logs and detailed logs. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

Enable log collection

You can use CloudLens for SLS to collect the important logs and detailed logs of Log Service. The operations that are performed to enable log collection for important logs are similar to the operations that are performed to enable log collection for detailed logs. In this example, enable log collection for detailed logs.

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click .
3. On the **SLS Cluster Access** tab of the **Access Management** page, find the project and click **Enable** in the **Detailed Logs** column.
4. In the **Enable Detailed Logs Collect** dialog box, select the project and click **Confirm**.

What to do next

After you enable , you can perform the following operations on the **Access Management** page.

Operation	Description

Operation	Description
Manage projects	After you enable , displays all projects that meet specified conditions within the current Alibaba Cloud account. Click the project that you want to manage. Then, you are navigated to the Project Overview page of the project.
Disable log collection	Find the project in which you want to disable log collection and click **Disable** in the Detailed Logs or Important Logs column.
View reports	Find the project whose reports you want to view and click **Query Reports** in the Actions column. Then, click the dashboard whose reports you want to view. Then, you are navigated to the dashboard page. For more information, see View data reports.
Reconfigure indexes	On the **Destination Logstore** tab, find the Logstore for which you want to reconfigure indexes and click **Reset** to reconfigure the indexes to the latest built-in version. **Notice** After the reconfiguration, the Logstore uses the latest built-in indexes, and custom indexes are deleted. The new indexes take effect only for new data in the Logstore.

7.4. View data reports

The application provides the following out-of-the-box dashboards: Access Traffic Monitoring, Access Exception Monitoring, Consumer Group Monitoring, Logtail Overall Status, Logtail File Collection Monitoring, Logtail Exception Monitoring, and Operation Monitoring. You can use the dashboards to monitor and analyze the consumption of resources in Log Service.

Prerequisites

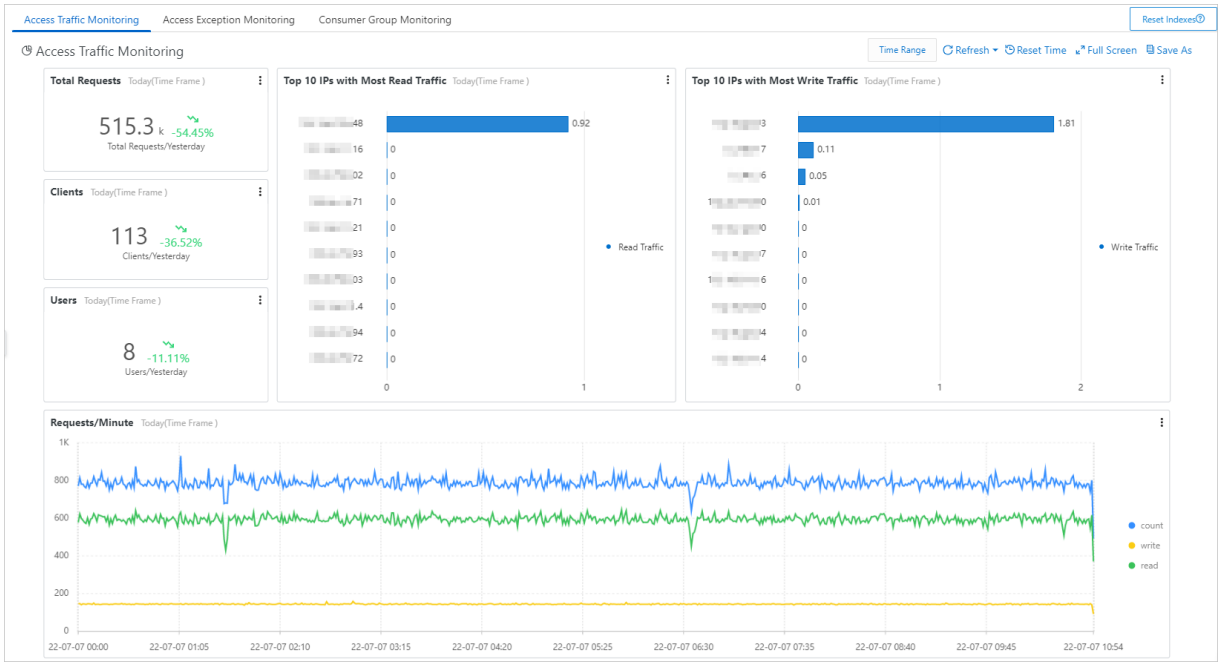
Log collection for important logs or detailed logs is enabled for a project. For more information, see Enable the log collection feature.

Entry point

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click .
3. In the left-side navigation pane, click **Report Center** and click Access Monitoring, Collection Monitoring, or Operation Monitoring.
4. In the upper-left corner of the page that appears, select the project whose reports you want to view.

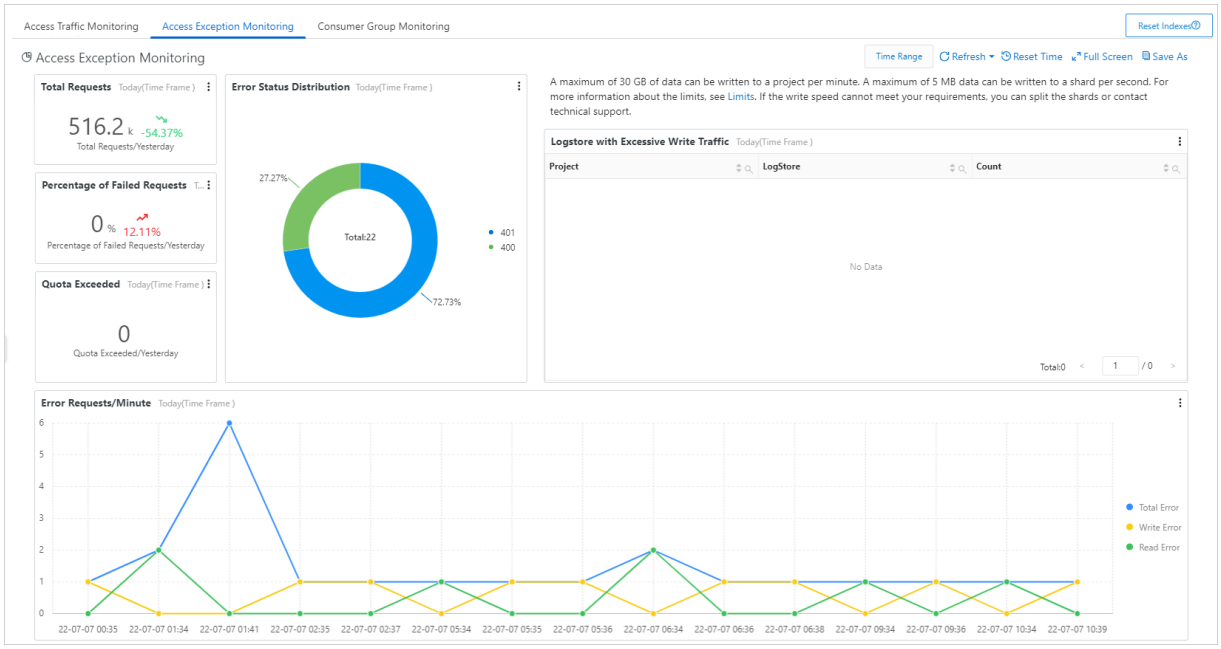
Access Traffic Monitoring

The **Access Traffic Monitoring** dashboard displays information about the access to Log Service in charts. The charts include Total Requests, Clients, Users, Top 10 IPs with Most Read Traffic, Top 10 IPs with Most Write Traffic, Requests, and Write Traffic.



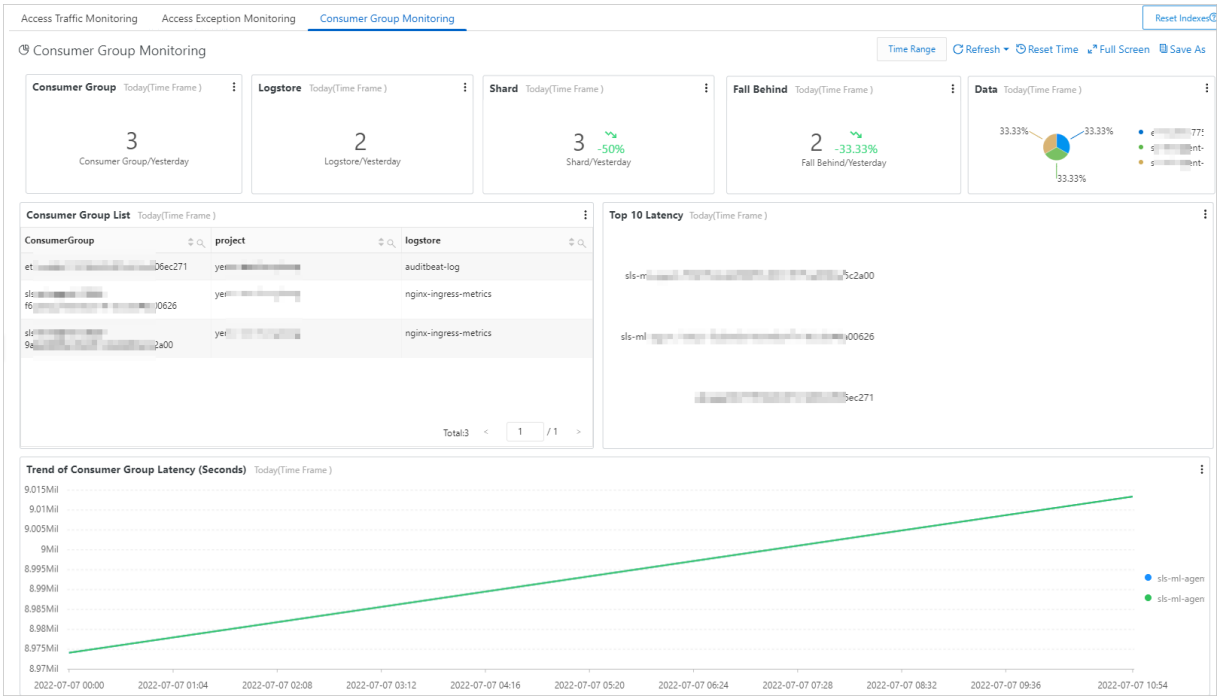
Access Exception Monitoring

The **Access Exception Monitoring** dashboard displays information about abnormal access to Log Service in charts. The charts include Total Requests, Percentage of Failed Requests, Quota Exceeded, Error Status Distribution, Logstore with Excessive Write Traffic, Error Requests, and Trend of Request Latency.



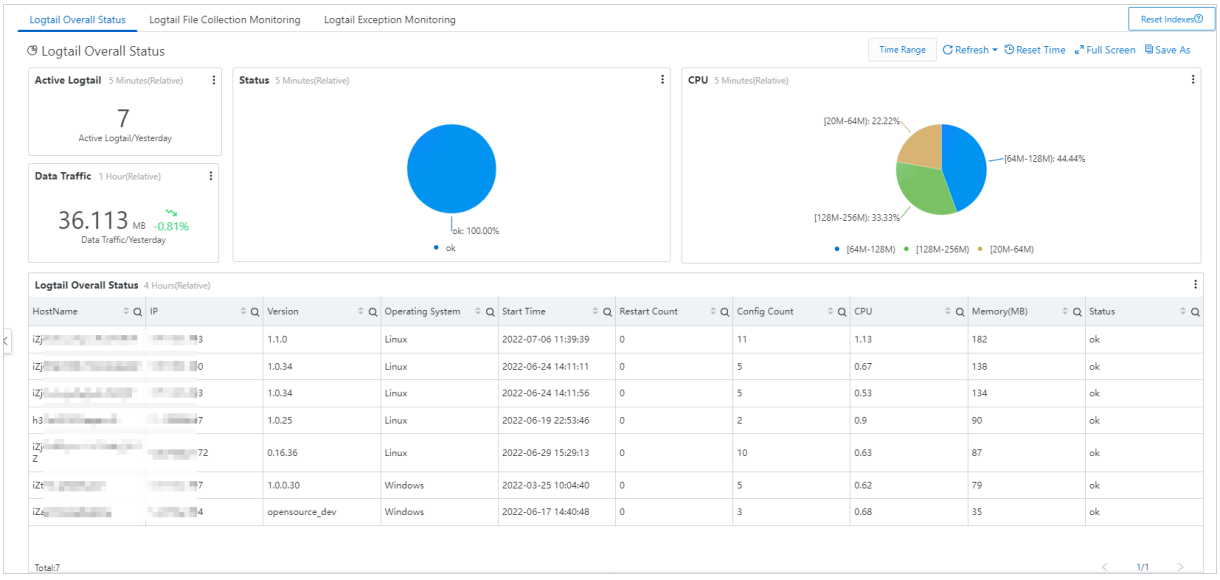
Consumer Group Monitoring

The **Consumer Group Monitoring** dashboard displays information about consumer groups in charts. The charts include Consumer Group, Logstore, Shard, Fall Behind, Data, Consumer Group List, Top 10 Latency, and Trend of Consumer Group Latency.



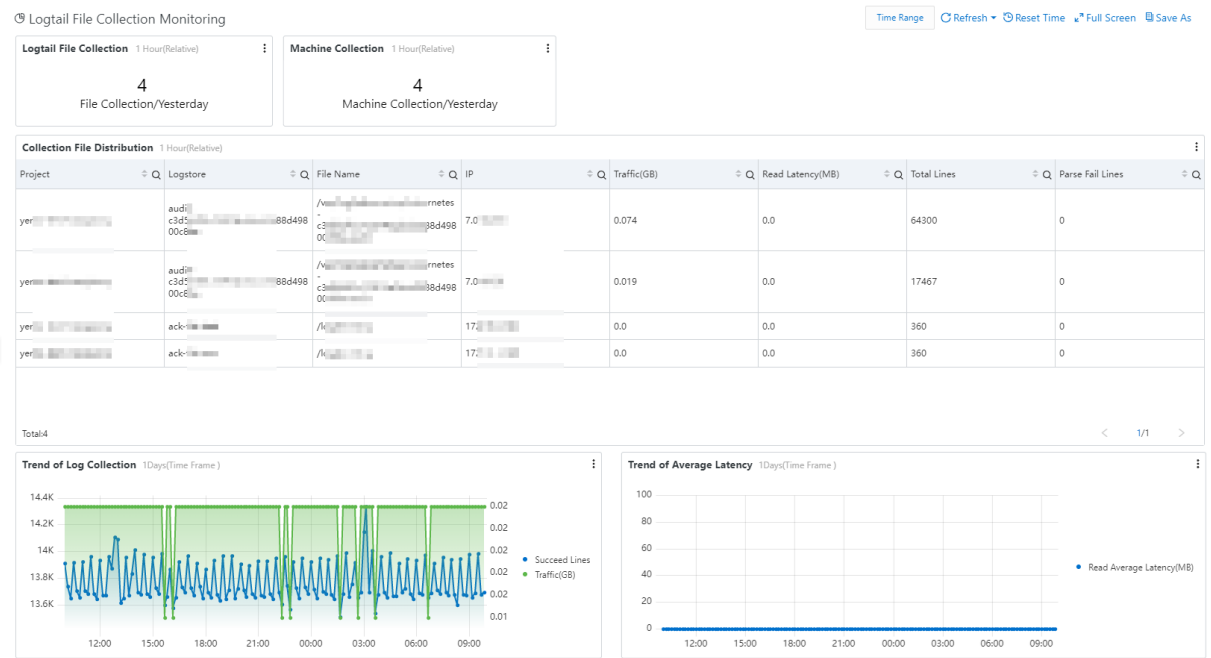
Logtail Overall Status

The **Logtail Overall Status** dashboard displays information about Logtail in charts. The charts include Active Logtail, Data Traffic, Status, CPU, Logtail Overall Status, Trend of CPU, Trend of Memory, and Data Sending Traffic.



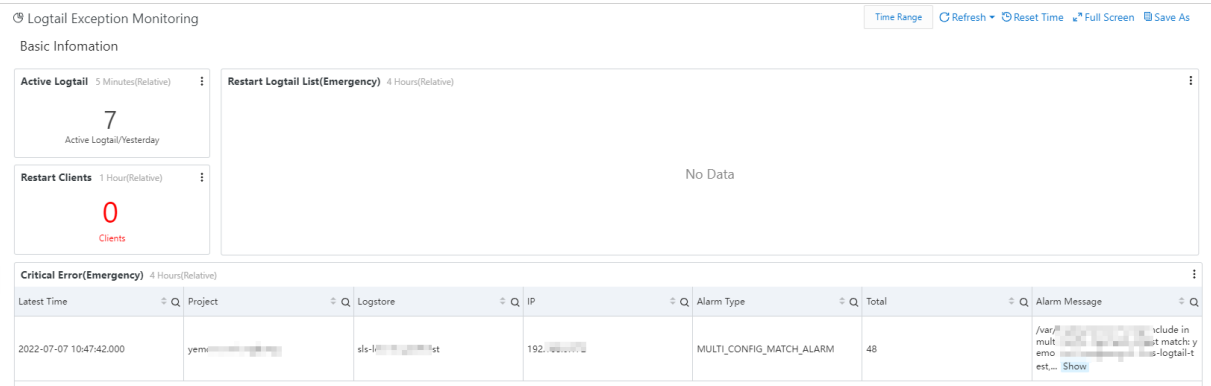
Logtail File Collection Monitoring

The **Logtail File Collection Monitoring** dashboard displays information about the files from which you want to collect logs in charts. The charts include Logtail File Collection, Machine Collection, Collection File Distribution, Trend of Log Collection, Trend of Average Latency, Trend of Parse Failure, and Trend of Send.



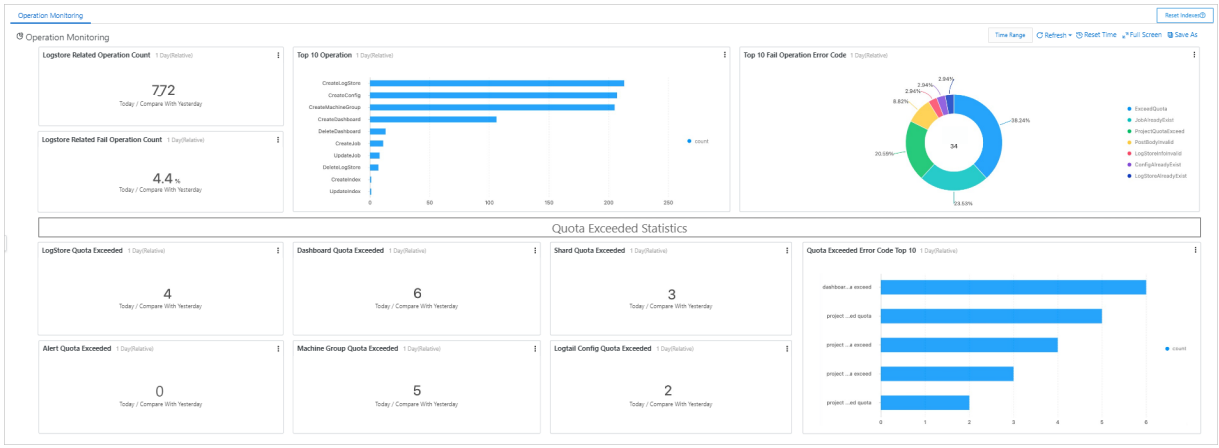
Logtail Exception Monitoring

The **Logtail Exception Monitoring** dashboard displays information about Logtail exceptions in charts. The charts include Active Logtail, Restart Logtail List, Restart Clients, and Critical Error.



Operation Monitoring

The **Operation Monitoring** dashboard displays information about operation records and the quotas that are exceeded in charts. The charts include Logstore Related Operation Count, Logstore Related Fail Operation Count, Top 10 Operation, Top 10 Fail Operation Error Code, LogStore Quota Exceeded, Dashboard Quota Exceeded, Shard Quota Exceeded, Alert Quota Exceeded, Machine Group Quota Exceeded, and Logtail Config Quota Exceeded.


Operation Monitoring
Operation Monitoring
Logstore Related Operation Count
772
Logstore Related Fail Operation Count
4.4%
Top 10 Operation
Top 10 Fail Operation Error Code
Quota Exceeded Statistics
LogStore Quota Exceeded
4
Dashboard Quota Exceeded
6
Shard Quota Exceeded
3
Alert Quota Exceeded
0
Machine Group Quota Exceeded
5
Logtail Config Quota Exceeded
2
Quota Exceeded Error Code Top 10

8.CloudLens for PolarDB

8.1. Usage notes

Log Service provides the CloudLens for PolarDB application. CloudLens for PolarDB allows you to manage PolarDB for MySQL clusters in a centralized manner and collect the slow query logs, error logs, audit logs, and metrics of the clusters. This topic describes the features, assets, billing, and limits of CloudLens for PolarDB.

Features

CloudLens for PolarDB provides the following features:

- Allows you to manage all PolarDB for MySQL clusters within your Alibaba Cloud account in a centralized manner.
- Allows you to enable the data collection feature for the slow query logs, error logs, audit logs, and metrics of PolarDB for MySQL clusters with a few clicks. You can also manage the data collection status of the clusters in a centralized manner.
- Allows you to store, query, and analyze the slow query logs, error logs, audit logs, and metrics of PolarDB for MySQL clusters in real time.
- Provides various reports.

Assets

- Logstore

  After you enable the data collection feature for a PolarDB for MySQL cluster in CloudLens for PolarDB, Log Service automatically creates a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-<Region ID> format and two Logstores named polardb_audit_log and polardb_log in the region where the cluster resides.

  - polardb_audit_log: This Logstore is used to store the audit logs of PolarDB for MySQL clusters.

    Audit logs are used to review operations and ensure security compliance.

    - Audit logs help security auditors obtain information such as operator identities and time of data modifications, and identify internal risks such as abuse of permissions and execution of non-compliant commands.
    - Audit logs help business systems meet audit requirements to ensure security compliance.

  - polardb_log: This Logstore is used to store the slow query logs and error logs of PolarDB for MySQL clusters.
    - Slow query logs record the requests whose execution time exceeds a specified threshold. You can use slow query logs to resolve performance issues and optimize requests.
    - Error logs record information about execution errors in databases. You can use error logs to identify issues.

- Metricstore

  The first time you enable the data collection feature for metrics, you must select a region. Then, Log Service creates a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-<Region ID> format and a Metricstore named polardb_metric in the region.

> **Note** Metrics are centrally stored in the region that you select when you enable the data collection feature. After you enable the data collection feature for metrics of other PolarDB for MySQL clusters, Log Service stores all collected metrics in the Metricstore that is created in the selected region. Example: aliyun-product-data-16****50-cn-hangzhou/polardb_metric.

- polardb_metric: This Metricstore is used to store the metrics of PolarDB for MySQL clusters.

  PolarDB for MySQL provides various metrics to monitor the running status of PolarDB for MySQL clusters.

Billing

- PolarDB for MySQL

  After you enable the data collection feature for the audit logs of a PolarDB for MySQL cluster, the SQL Explorer feature is automatically enabled in the cluster. You are charged for the SQL Explorer feature based on the storage space that is occupied by audit logs. For more information, see Billable items.

- Log Service

  After Log Service collects logs from PolarDB for MySQL, you can transform or ship the logs. You can also read streaming data over the Internet. You are charged for data transformation, data shipping, and read traffic over the Internet. The charges are included into your Log Service bills. For more information, see Billable items.

Limits

- You can collect only the logs and metrics of PolarDB for MySQL clusters.
- The collection feature for audit logs in CloudLens for PolarDB depends on the SQL Explorer feature of PolarDB for MySQL.

  If the SQL Explorer feature is disabled, CloudLens for PolarDB automatically enables the feature.

- The following table describes the regions in which CloudLens for PolarDB is supported.

Cloud type	Region
Alibaba Cloud public cloud	China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Hangzhou), China (Shanghai), China (Shenzhen), China (Chengdu), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Japan (Tokyo), India (Mumbai), US (Silicon Valley), US (Virginia), Germany (Frankfurt), and UK (London)

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

Notice If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

8.2. Grant the operation permissions on CloudLens for PolarDB to a RAM user

This topic describes how to grant the operation permissions on CloudLens for PolarDB to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant the operation permissions on CloudLens for PolarDB to a RAM user in one of the following modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode requires complex configurations and provides fine-grained access control.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the text editor with one of the following scripts, and then click **Next Step**.

   You can grant the read-only permissions or read and write permissions on CloudLens for PolarDB to a RAM user.

- Read-only permissions: Use the following script to authorize the RAM user only to view each page of CloudLens for PolarDB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "log:GetProductDataCollection",
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:polardb:*:*:dbcluster/*"
            ],
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

- Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by CloudLens for PolarDB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
```

```
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:GetProductDataCollection",
                "log:OpenProductDataCollection",
                "log:CloseProductDataCollection"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:polardb:*:*:dbcluster/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.polardb.global_conf
.standard_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
                    "ram:ServiceName": "audit.log.aliyuncs.com",
                    "ram:ServiceName": "polardb.aliyuncs.com"
                }
            }
        }
    ],
    "Version": "1"
}
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the policy name to log-polardb-policy.

3. Attach the policy to the RAM user.
   i. In the left-side navigation pane, choose **Identities > Users**.
   ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.
   iii. In the **Add Permissions** panel, go to the **Select Policy** section, click **Custom Policy**, and then click the policy that you create in Step . In this example, click log-polardb-policy.
   iv. Click **OK**.

8.3. Enable data collection

CloudLens for PolarDB allows you to enable data collection for the audit logs, slow query logs, error logs, and metrics of PolarDB for MySQL clusters with a few clicks. This topic describes how to enable data collection in CloudLens for PolarDB. This topic also describes the operations that you can perform after you enable data collection.

Prerequisites

A PolarDB for MySQL cluster is created. For more information, see Purchase a pay-as-you-go cluster or Purchase a subscription cluster.

Initially configure CloudLens for PolarDB

> **Notice** You need to perform this operation only once.

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab and click **CloudLens for PolarDB**.
3. Enable CloudLens for PolarDB by following the on-screen instructions.

   When you enable CloudLens for PolarDB, the system automatically authorizes CloudLens for PolarDB to assume the service-linked role AliyunServiceRolePolicyForSLSAudit. This allows CloudLens for PolarDB to collect logs and metrics from PolarDB for MySQL. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

Enable data collection

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab and click **CloudLens for PolarDB**.
3. On the **PolarDB Cluster Access** tab of the **Access Management** page, find the PolarDB for MySQL cluster and click **Enable**.

   Log Service allows you to collect the audit logs, slow query logs, error logs, and metrics of a PolarDB for MySQL cluster. You can enable data collection based on your business requirements.

> **Notice** After you enable data collection for the audit logs of a PolarDB for MySQL cluster, the SQL Explorer feature is automatically enabled in the cluster. You are charged for the SQL Explorer feature based on the amount of storage that is occupied by audit logs. For more information, see Billable items.

4. In the dialog box that appears, click **Confirm**.

   Metrics are centrally stored. The collected metrics of all PolarDB for MySQL clusters within the current Alibaba Cloud account are stored in the same Metricstore. The first time you enable data collection for metrics, you must select a region in which you want to store the collected metrics. For more information, see Assets.

What to do next

After you enable CloudLens for PolarDB, you can perform the following operations on the **Access Management** page.

Operation	Description
Manage PolarDB for MySQL clusters	After you enable CloudLens for PolarDB, CloudLens for PolarDB displays all PolarDB for MySQL clusters within your Alibaba Cloud account. Click a PolarDB for MySQL cluster to go to the Apsara PolarDB console. You can view the details about the cluster. You can also perform other operations. For example, you can log on to a database or migrate a database. For more information, see Feature list.
Disable data collection	Find a PolarDB for MySQL cluster and click **Disable** to disable data collection. **Notice** If you disable data collection for audit logs, the SQL Explorer feature is not automatically disabled. If you no longer need to collect the audit logs of a PolarDB for MySQL cluster, you can disable the SQL Explorer feature in the Apsara PolarDB console. For more information, see Disable the SQL Explorer feature.
Query and analyze data	• Find a PolarDB for MySQL cluster, click **Log Query**, and then select the type of log that you want to query and analyze to go to the Logstore in which the logs are stored. Then, you can view the raw logs and perform query and analysis on the logs. For more information, see Query and analyze logs. • Find a PolarDB for MySQL cluster, click **Log Query**, and then select **Performance Metrics** to go to the Metricstore in which the metrics are stored. Then, you can view the raw data of the metrics and perform query and analysis on the metrics. For more information, see Query and analyze time series data.
Modify data retention periods	On the **Destination Logstore** tab, find a Logstore or a Metricstore and click the icon to modify the retention period of data in the Logstore or Metricstore.

What's next

View performance monitoring dashboards

8.4. View performance monitoring dashboards

CloudLens for PolarDB provides performance monitoring dashboards that display the metrics of PolarDB for MySQL clusters in real time. You can view the dashboards to check the running status of PolarDB for MySQL clusters and obtain detailed monitoring data. The data helps you locate O&M issues in an efficient manner.

Prerequisites

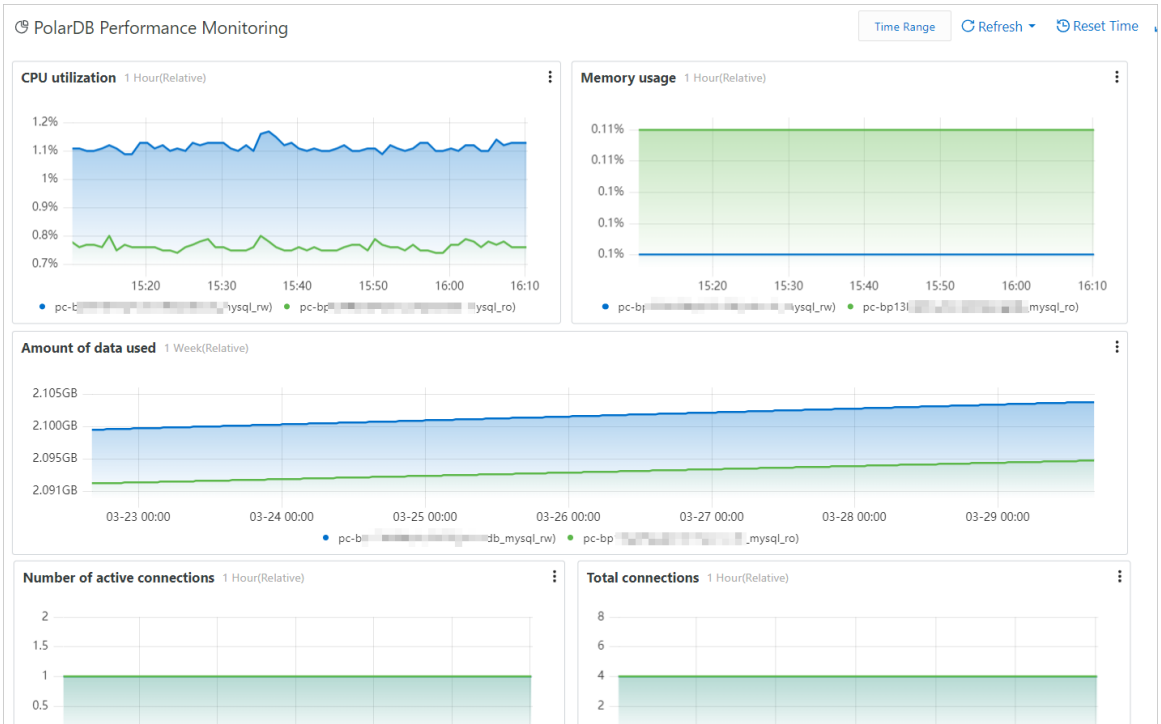
The data collection feature is enabled for the metrics of a PolarDB for MySQL cluster. For more information, see Enable data collection.

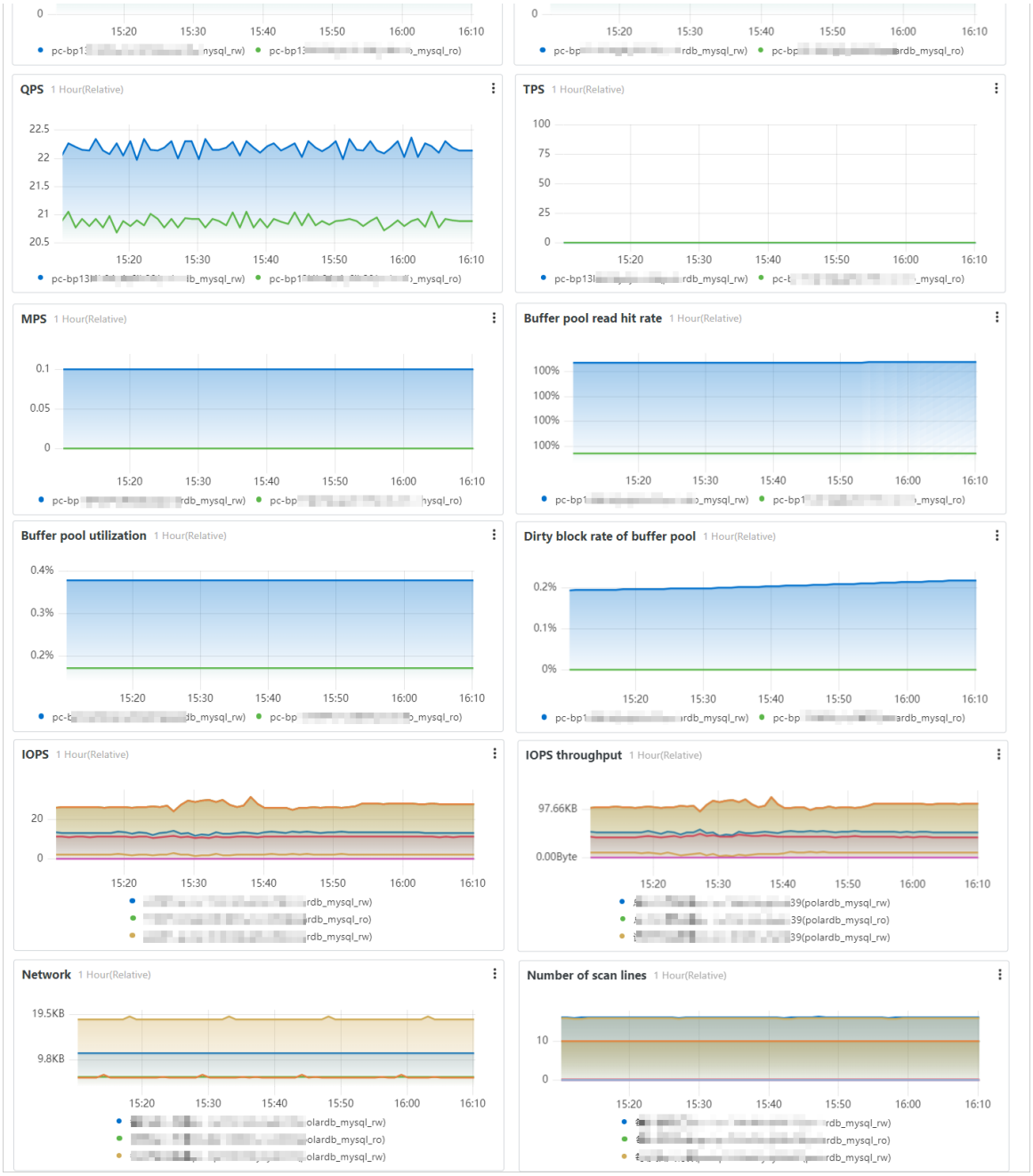
Entry point

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab and click **CloudLens for PolarDB**.
3. In the left-side navigation pane, click **Performance Monitoring**.

Data details

On the **Performance Monitoring** page, you can view the metrics of a PolarDB for MySQL cluster within a specified time range. The metrics include the CPU utilization, memory usage, used data volume, number of active connections, total number of connections, queries per second (QPS), transactions per second (TPS), MPS, read hit ratio of the buffer pool, buffer pool utilization, dirty ratio of the buffer pool, and IOPS.




QPS 1 Hour(Relative)
TPS 1 Hour(Relative)
MPS 1 Hour(Relative)
Buffer pool read hit rate 1 Hour(Relative)
Buffer pool utilization 1 Hour(Relative)
Dirty block rate of buffer pool 1 Hour(Relative)
IOPS 1 Hour(Relative)
IOPS throughput 1 Hour(Relative)
Network 1 Hour(Relative)
Number of scan lines 1 Hour(Relative)
15:20
15:30
15:40
15:50
16:00
16:10

9.CloudLens for Redis

9.1. Usage notes

Log Service and ApsaraDB for Redis jointly launch the CloudLens for Redis application. You can use the application to manage ApsaraDB for Redis instances in a centralized manner and collect the run logs, slow query logs, and audit logs of the instances. This topic describes the features, assets, billing, and limits of CloudLens for Redis.

Features

CloudLens for Redis provides the following features:

- Allows you to manage all ApsaraDB for Redis instances that meet specified conditions within your Alibaba Cloud account in a centralized manner. For more information about the conditions, see Limits.
- Allows you to enable the log collection feature for run logs, slow query logs, and audit logs of ApsaraDB for Redis instances with a few clicks and manage the log collection status of the instances in a centralized manner.
- Allows you to store, query, and analyze the run logs, slow query logs, and audit logs of ApsaraDB for Redis instances in real time.
- Provides various reports.

Assets

After you enable CloudLens for Redis and data collection for an ApsaraDB for Redis instance, Log Service automatically creates a project named in the nosql-Alibaba Cloud account ID-Region ID format and two Logstores named redis_audit_log_standard and redis_slow_run_log in the region where the instance resides.

- redis_audit_log_standard: This Logstore is used to store the audit logs of ApsaraDB for Redis instances.

  Audit logs are used to review operations and ensure security compliance.

  - Audit logs help security auditors obtain information such as operator identities and time of data modification, and identify internal risks such as abuse of permissions and execution of non-compliant commands.
  - Audit logs help business systems meet audit requirements to ensure security compliance.

- redis_slow_run_log: This Logstore is used to store the slow query logs and run logs of ApsaraDB for Redis instances.
  - Slow query logs record the requests whose execution time exceeds a specified threshold. You can use slow query logs to resolve performance issues and optimize requests.
  - Run logs record the status of ApsaraDB for Redis instances during their lifecycles. You can use run logs to resolve O&M issues.

Billing

- ApsaraDB for Redis: You are charged based on the storage space and retention period of audit logs. The fees that you must pay vary based on the region of your ApsaraDB for Redis instance. For more information, see Billable items and prices.

  You are not charged when you write, store, or query slow query logs and run logs.

- Log Service: After you collect logs from ApsaraDB for Redis to Log Service, you can transform the logs or ship the logs. You can also read the logs in stream mode by using a public endpoint of Log Service. You are charged for the read traffic consumed when you access Log Service over the Internet, data transformation, and data shipping. For more information, see Billable items.

Limits

- Log Service collects logs from an ApsaraDB for Redis instance only if the instance meets the following conditions:
  - The audit logging feature is enabled for the instance. For more information, see Enable the new audit log feature.
  - The major engine version of the instance is Redis 4.0 or later, and the minor engine version of the instance is the latest version. For more information about how to upgrade the engine versions of an ApsaraDB for Redis instance, see Upgrade the major version and Update the minor version.
  - The instance runs Community Edition or is a performance-enhanced instance that uses local disks.
- You cannot disable log collection for slow query logs or run logs.

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

> **Notice** If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

9.2. Grant the operation permissions on CloudLens for Redis to a RAM user

This topic describes how to grant the operation permissions on CloudLens for Redis to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant the operation permissions on CloudLens for Redis to a RAM user in one of the following modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode requires complex configurations and provides fine-grained access control.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the text editor with one of the following scripts, and then click **Next Step**.

   You can grant the read-only permissions or read and write permissions on CloudLens for Redis to a RAM user.

   - Read-only permissions: Use the following script to authorize the RAM user only to view each page of CloudLens for Redis.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:PassRole",
            "Resource": "acs:ram:*:*:role/aliyunserviceroleforslsaudit",
            "Effect": "Allow"
        },
        {
            "Action": "log:GetProductDataCollection",
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:kvstore:*:*:instance/*"
            ],
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

- Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by CloudLens for Redis.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
```

```
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:PassRole",
            "Resource": "acs:ram:*:*:role/aliyunserviceroleforslsaudit",
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:GetProductDataCollection",
                "log:OpenProductDataCollection",
                "log:CloseProductDataCollection"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:kvstore:*:*:instance/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.redis.global_conf.s
tandard_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
                    "ram:ServiceName": "audit.log.aliyuncs.com",
                    "ram:ServiceName": "r-kvstore.aliyuncs.com"
                }
            }
```

```
                }
            }
        ],
        "Version": "1"
    }
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the policy name to log-redis-policy.

3. Attach the policy to the RAM user.
   i. In the left-side navigation pane, choose **Identities > Users**.
   ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.
   iii. In the **Add Permissions** panel, go to the **Select Policy** section, click **Custom Policy**, and then click the policy that you create in Step . In this example, click log-redis-policy.
   iv. Click **OK**.

9.3. Enable log collection

CloudLens for Redis allows you to enable log collection for the slow query logs, operational logs, and audit logs of ApsaraDB for Redis instances with a few clicks. This topic describes how to enable log collection in CloudLens for Redis. This topic also describes the operations that you can perform after you enable log collection.

Prerequisites

An ApsaraDB for Redis instance that meets the following conditions is created. For more information, see Step 1: Create an ApsaraDB for Redis instance.

- The major engine version of the instance is Redis 4.0 or later, and the minor engine version of the instance is up to date. For more information about how to upgrade the engine versions of an ApsaraDB for Redis instance, see Upgrade the major version and Update the minor version.
- The instance runs Community Edition or is a performance-enhanced instance that uses local disks.

Initially configure CloudLens for Redis

Notice You need to perform this operation only once.

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab and click **CloudLens for Redis**.
3. Enable CloudLens for Redis by following the on-screen instructions.

   When you enable CloudLens for Redis, the system automatically completes the following authorization:

   - Authorizes CloudLens for Redis to assume the service-linked role AliyunServiceRolePolicyForSLSAudit to collect logs from ApsaraDB for Redis. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.
   - Authorizes ApsaraDB for Redis to assume the service-linked role AliyunServiceRoleForKvstore to access resources in Log Service. For more information, see Service linked roles in ApsaraDB for Redis.

Enable log collection

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab and click **CloudLens for Redis**.
3. On the **Redis Cluster Access** tab of the **Access Management** page, find the ApsaraDB for Redis instance and click **Enable**.

   > **Notice** After you click **Enable**, the log collection feature is enabled for the audit logs, slow query logs, and operational logs of the ApsaraDB for Redis instance. You can disable the log collection feature only for audit logs. However, you cannot disable the feature for slow query logs or operational logs.

4. In the **Enable Audit Logs Collect** dialog box, configure the **Retention Period** parameter and click **Confirm**.

   Valid values of **Retention Period**: 1 to 365.

What to do next

After you enable CloudLens for Redis, you can perform the following operations on the **Access Management** page.

Operation	Description
Manage ApsaraDB for Redis instances	After you enable CloudLens for Redis, CloudLens for Redis displays all ApsaraDB for Redis instances that meet the specified conditions within your Alibaba Cloud account. Click an ApsaraDB for Redis instance to go to the ApsaraDB for Redis console. You can view the details about the instance. You can also perform other operations. For example, you can log on to a database or migrate a database. For more information, see Manage ApsaraDB for Redis instances.
Disable log collection	Find an ApsaraDB for Redis instance and click **Disable** to disable log collection for audit logs.
Query and analyze logs	Find an ApsaraDB for Redis instance, click **Index Search**, and then select the type of log that you want to query and analyze to go to the Logstore in which the logs are stored. You can view the raw logs and perform query and analysis on the logs. For more information, see Query and analyze logs.

What's next

View analysis dashboards

9.4. View data reports

CloudLens for Redis provides various charts that display the analysis results for the logs of ApsaraDB for Redis instances.

Prerequisites

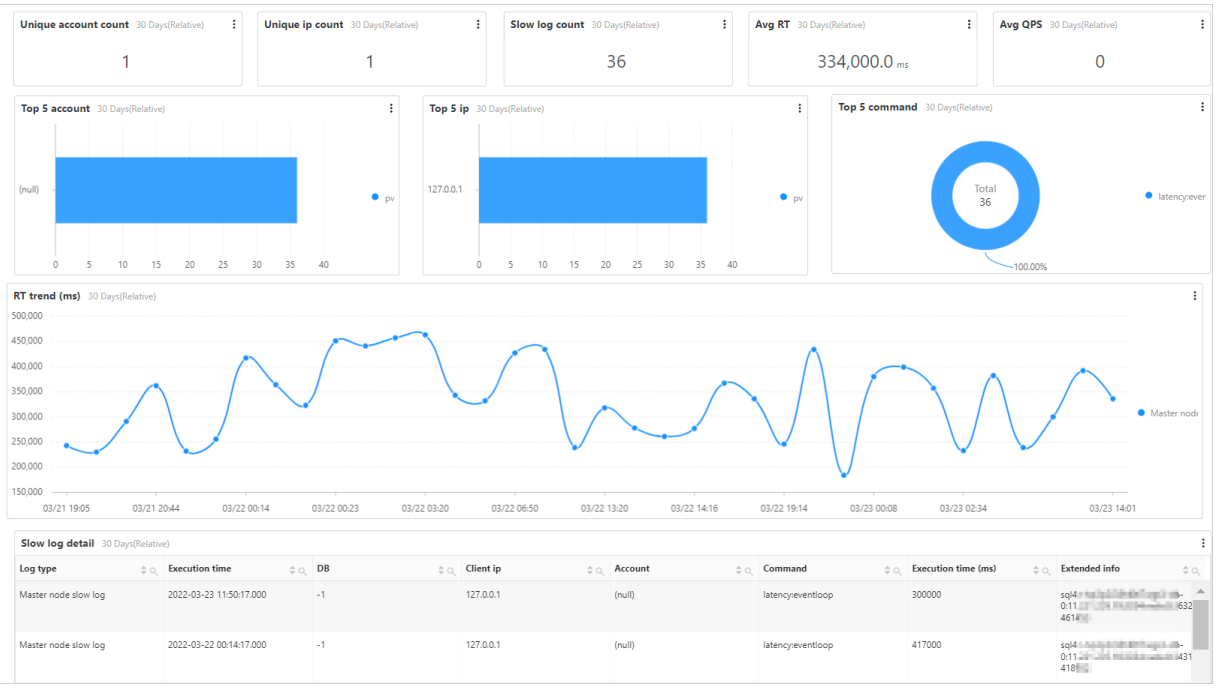
Log collection is enabled. For more information, see Enable log collection.

Entry point

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click **CloudLens for Redis**.
3. In the left-side navigation pane, click **Report Center**.

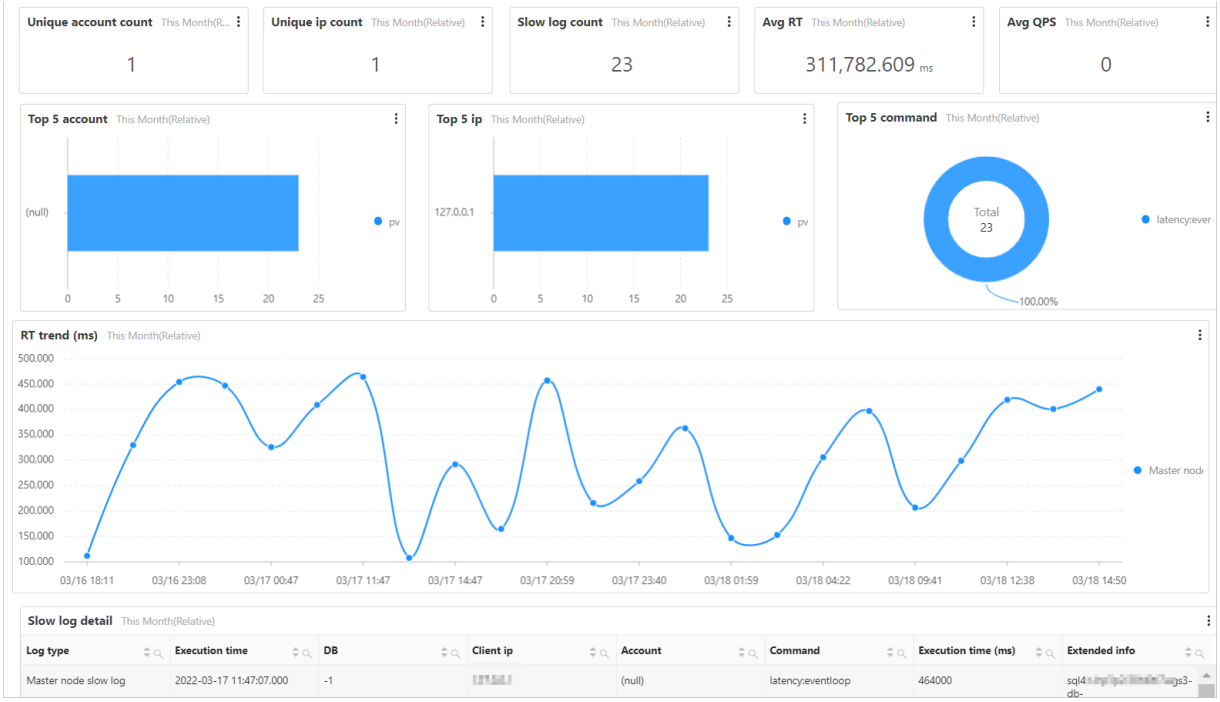
Slow Query Log Center

The **Slow Query Log Center** dashboard displays the analysis results of slow query logs, including the number of access users, the number of access clients, the number of slow query logs, the average response time, and the average queries per second (QPS).



Audit Log Center

The **Audit Log Center** dashboard displays the analysis results of audit logs, including the number of access users, the number of access clients, the number of audit logs, the average response time, and the average QPS.


Unique account count This Month(R...
1
Unique ip count This Month(Relative)
1
Slow log count This Month(Relative)
23
Avg RT This Month(Relative)
311,782.609 ms
Avg QPS This Month(Relative)
0
Top 5 account This Month(Relative)
(null)
pv
0 5 10 15 20 25
Top 5 ip This Month(Relative)
127.0.0.1
pv
0 5 10 15 20 25
Top 5 command This Month(Relative)
Total 23
100.00%
latency:ever
RT trend (ms) This Month(Relative)
500,000
450,000
400,000
350,000
300,000
250,000
200,000
150,000
100,000
03/16 18:11
03/16 23:08
03/17 00:47
03/17 11:47
03/17 14:47
03/17 20:59
03/17 23:40
03/18 01:59
03/18 04:22
03/18 09:41
03/18 12:38
03/18 14:50
Master nod
Slow log detail This Month(Relative)
Log type
Execution time
DB
Client ip
Account
Command
Execution time (ms)
Extended info
Master node slow log
2022-03-17 11:47:07.000
-1
(null)
latency:eventloop
464000

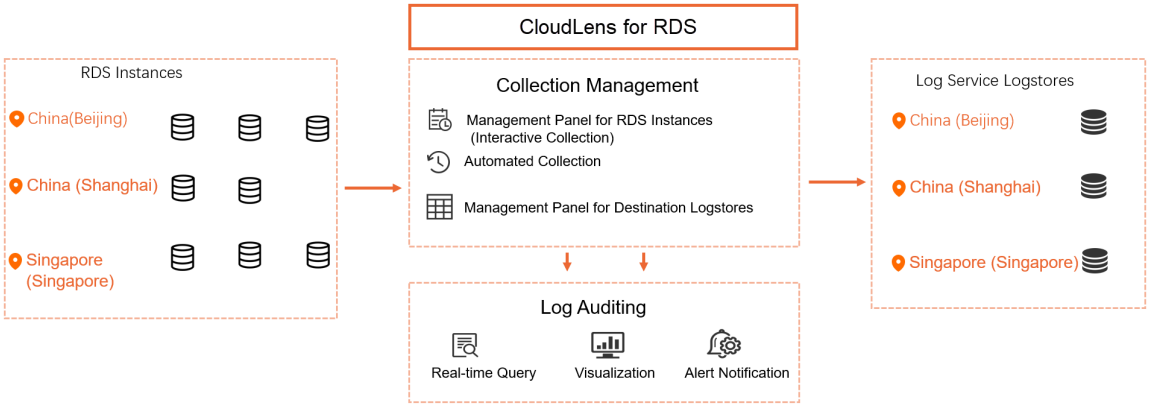
10.CloudLens for RDS

10.1. Usage notes

Log Service provides the CloudLens for RDS application. You can use the application to check the collection status of SQL audit logs for ApsaraDB RDS instances in real time and manage collection configurations in a centralized manner. You can also audit and analyze collected logs and configure alerts for the logs.

Features

CloudLens for RDS provides the following features:



- Collection management
  - Allows you to manage the collection status of SQL audit logs for ApsaraDB RDS instances in a centralized manner.
  - Automatically collects SQL audit logs from existing ApsaraDB RDS instances and new instances.
  - Allows you to manage projects and Logstores in a centralized manner.
- Log audit
  - Allows you to store, query, and analyze SQL audit logs of ApsaraDB RDS instances in real time.
  - Provides various reports. You can subscribe to these reports and configure settings to receive the reports by using emails or DingTalk group messages.
  - Provides various built-in alert rules, supports flexible configurations for alert policies, and sends alert messages in a timely and accurate manner.

Supported log types

The SQL audit logs of an ApsaraDB RDS database record all operations that are performed on the database. The logs are obtained by the system based on network protocol analysis, which consumes only a small amount of CPU resources and does not affect the execution of SQL statements. The SQL audit logs record the following operations and related information:

- Database logons and logoffs.
- DDL operations: SQL statements that define a database structure. Examples: CREATE, ALTER DROP, TRUNCATE, and COMMENT.
- DML operations: SQL statements that perform specific operations. Examples: SELECT, INSERT, UPDATE, and DELETE.

- Other operations that are performed by executing SQL statements. Examples: rollback and control.
- The execution latency, execution results, and number of affected rows of SQL statements.

Assets

- Custom projects and Logstores

  > **Notice** Do not delete the projects or Logstores that are used for the SQL audit logs shipped from ApsaraDB RDS. Otherwise, subsequent logs cannot be shipped to Log Service.

- Dedicated dashboards

  By default, Log Service generates three dashboards for the feature.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because the dashboards may be upgraded or updated at all times. You can create a custom dashboard to visualize query results. For more information, see Create a dashboard.

Dashboard	Description
RDS Operation Center	Displays statistics about access to databases and active databases. The statistics include the number of databases on which the operations are performed, number of tables on which the operations are performed, and number of execution errors. The statistics also include the total number of inserted rows, total number of updated rows, total number of deleted rows, and total number of obtained rows.
RDS Performance Center	Displays the metrics that are related to O&M reliability. The metrics include the peak bandwidth for all SQL statements that are executed, peak bandwidth for SQL statements that query data, peak bandwidth for SQL statements that insert data, peak bandwidth for SQL statements that update data, and peak bandwidth for SQL statements that delete data. The metrics also include the average execution time of all SQL statements, average execution time of SQL statements that query data, average execution time of SQL statements that update data, and average execution time of SQL statements that delete data.
RDS Security Center	Displays the metrics that are related to database security. The metrics include the number of errors, number of logon failures, number of bulk deletion events, number of bulk modification events, and number of times that risky SQL statements are executed. The metrics also include the distribution of error operations by type, distribution of clients that have errors on the Internet, and clients that have the largest number of errors.

Billing

- The log collection feature of CloudLens for RDS depends on the SQL Explorer feature of ApsaraDB RDS for MySQL. The charges that are incurred by the SQL Explorer feature are included into your ApsaraDB RDS bills. For more information, see Billable items, billing methods, and pricing.

> **Note** If your ApsaraDB RDS for MySQL instance runs RDS Enterprise Edition, you are not charged for the SQL Explorer feature.

- After you use Log Service to collect the SQL audit logs of ApsaraDB RDS instances, you are charged for data storage, read traffic, requests, data transformation, and data shipping. For more information, see Pay-as-you-go.

Limits

- Log Service can collect SQL audit logs only from the following types of ApsaraDB RDS instances:

  ApsaraDB RDS for MySQL instances: All available RDS editions are supported, except RDS Basic Edition.
- The log collection feature of CloudLens for RDS depends on the SQL Explorer feature of ApsaraDB RDS for MySQL.

  After you enable the log collection feature for ApsaraDB RDS for MySQL instances in CloudLens for RDS, the system automatically enables the SQL Explorer feature of the ApsaraDB RDS for MySQL instances.
- The Log Service project that is used to store SQL audit logs collected from an ApsaraDB RDS instance must reside in the same region as the instance.
- All regions are supported, except Local Regions.

Log collection methods

Log Service can collect SQL audit logs from ApsaraDB RDS instances by using one of the following methods:

> **Note** If SQL audit logs are collected by using Method 1 or Method 3, you can apply the collection configurations that you create for one method to the other method. If SQL audit logs are collected by using Method 2, you cannot use the collection configurations that you create for Method 1 or Method 3. You must separately create collection configurations.

- Method 1: CloudLens for RDS
  - To collect SQL audit logs by using Method 1, log on to the Log Service console. In the **Log Application** section, click **CloudLens for RDS**.
  - If you want to collect SQL audit logs from ApsaraDB RDS instances that belong to the same Alibaba Cloud account, we recommend that you use this method.
- Method 2: Log Audit Service
  - To collect SQL audit logs by using Method 2, log on to the Log Service console. In the **Log Application** section, click **Log Audit Service**.
  - If you want to collect SQL audit logs from ApsaraDB RDS instances across Alibaba Cloud accounts or regions, we recommend that you use this method.
- Method 3: Import Data - RDS SQL Audit
  - To collect SQL audit logs by using Method 3, log on to the Log Service console. In the **Import Data** section, click **RDS SQL Audit - Cloud Products**.
  - This method is an alternative to Method 1.

Attribute	Import Data - RDS SQL Audit	Method 1: CloudLens for RDS	Log Audit Service
Specify an ApsaraDB RDS instance to collect logs	Supported	Supported	Supported
Specify a Logstore to store logs	Supported	Supported	Not supported
Collect SQL audit logs from ApsaraDB RDS instances across regions	Not supported	Not supported	Supported
Collect SQL audit logs from ApsaraDB RDS instances across Alibaba Cloud accounts	Not supported	Not supported	Supported
Automatic collection	Not supported	Supported	Supported
Manual collection	Supported	Supported	Not supported
View collection status in dashboards	Not supported	Supported	Not supported

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

Notice If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

10.2. Grant operation permissions to a RAM user

This topic describes how to grant the operation permissions on CloudLens for RDS to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant the operation permissions on CloudLens for RDS to a RAM user in one of the following

modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode allows you to perform fine-grained access control. However, the configurations in this mode are complex.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the text editor with one of the following scripts, and then click **Next Step**.

   You can grant the RAM user the read-only permissions or read and write permissions on CloudLens for RDS.

   - Read-only permissions: Use the following script to authorize the RAM user only to view each page of CloudLens for RDS.

```
{
    "Version": "1",
    "Statement": [
        {
            "Action": [
                "rds:DescribeSqlLogInstances",
                "rds:DisableSqlLogDistribution"
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard"
            ],
            "Resource": [
                "acs:log:*:*:project/sls-alert-*/logstore/*"
```

```
                "acs:log:*:*:project/sls-alert-*/dashboard/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:CreateProject"
            ],
            "Resource": [
                "acs:log:*:*:project/sls-alert-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ]
        },
        {
            "Action": [
                "ram:GetRole"
            ],
            "Resource": "acs:ram:*:*:role/aliyunlogarchiverole",
            "Effect": "Allow"
        }
    ]
}
```

- Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by CloudLens for RDS.

```
{
    "Version": "1",
    "Statement": [
        {
            "Action": [
                "rds:DescribeSqlLogInstances",
                "rds:DisableSqlLogDistribution",
                "rds:DisableSqlLogDistribution",
                "rds:EnableSqlLogDistribution",
                "rds:ModifySQLCollectorPolicy"
```

```
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard"
            ],
            "Resource": [
                "acs:log:*:*:project/sls-alert-*/logstore/*",
                "acs:log:*:*:project/sls-alert-*/dashboard/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:CreateProject"
            ],
            "Resource": [
                "acs:log:*:*:project/sls-alert-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
```

```
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ]
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.rds.global_conf.sin
gle_account_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
                    "ram:ServiceName": "audit.log.aliyuncs.com"
                }
            }
        },
        {
            "Action": [
                "ram:*"
            ],
            "Resource": [
                "acs:ram:*:*:role/aliyunlogarchiverole",
                "acs:ram:*:*:policy/AliyunLogArchiveRolePolicy"
            ],
            "Effect": "Allow"
        }
    ]
}
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the Name parameter to log-rds-policy.

3. Attach the policy to the RAM user.

i. In the left-side navigation pane, choose **Identities > Users**.

ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.

iii. In the **Select Policy** section of the **Add Permissions** panel, click **Custom Policy**. Then, click the policy that you created in Step .

iv. Click **OK**.

10.3. Enable the log collection feature

You can manually enable the log collection feature or configure the automatic collection feature in the CloudLens for RDS application. You can manually enable the log collection feature for only one RDS instance at a time. You can configure the automatic collection feature for multiple RDS instances. After you configure the automatic collection feature, Log Service automatically collects the audit logs of the existing and new RDS instances that meet the specified conditions. This topic describes how to enable the log collection feature in the CloudLens for RDS application.

Prerequisites

- If you want to manually enable the log collection feature for an RDS instance, you must create a Log Service project and a Logstore in the region where the RDS instance resides. For more information, see Create a project and a Logstore.
- If you use a RAM user, you must grant the RAM user the permissions to manage the CloudLens for RDS application. For more information, see Grant operation permissions to a RAM user.

Initial configurations

> **Notice**
>
> - The Alibaba Cloud account that you use to complete authorization must have the AliyunRamFullAccess permission.
> - You need to perform this operation only once.

1.
2. In the **Log Application** section, click **CloudLens for RDS**.
3. Grant permissions to the AliyunLogArchiveRole role as prompted.

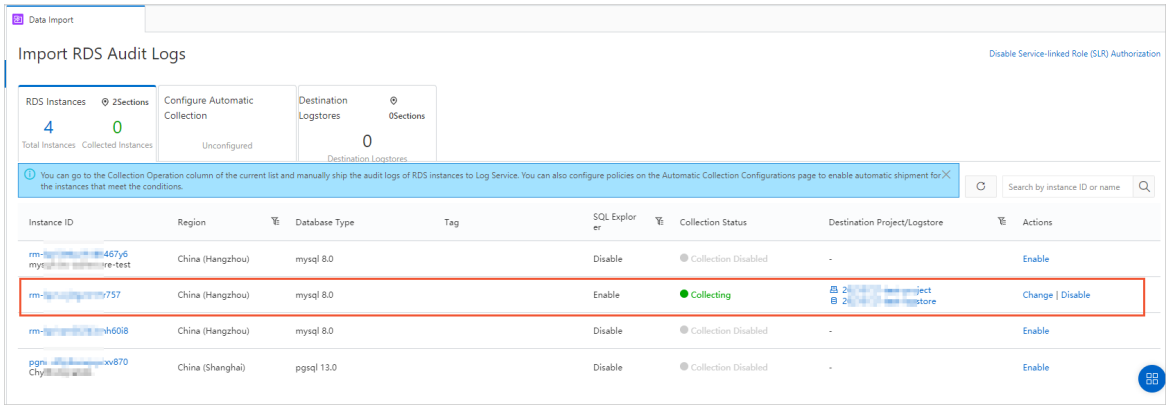
   After you perform this operation, Alibaba Cloud automatically creates a system role named AliyunLogArchiveRole that the CloudLens for RDS application can assume to access the resources of other Alibaba Cloud services.
4. Grant permissions to the AliyunServiceRoleForSLSAudit role as prompted.

   After you perform this operation, Alibaba Cloud automatically creates a service-linked role named AliyunServiceRoleForSLSAuditRDS that the CloudLens for RDS application can assume to collect RDS audit logs. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

Manually enable the log collection feature for an RDS instance

1.
2. In the **Log Application** section, click **CloudLens for RDS**.
3. On the **Data Import** tab, find the RDS instance that you want to manage. Then, click **Enable** in the Actions column.
4. In the **Select Destination** dialog box, select a destination project and a destination Logstore. Then, click **OK**.

   After the log collection feature is enabled, Log Service collects the audit logs of the RDS instance.



Configure automatic collection

1.
2. In the **Log Application** section, click **CloudLens for RDS**.
3. On the **Data Import** tab, click **Configure Automatic Collection**.
4. Click the icon.
5. Specify conditions for log collection.

   You can select Alibaba Cloud account ID, Region, Instance ID, Instance Name, Database Type, Database Version, or Tag from the Object drop-down list and then specify a condition.

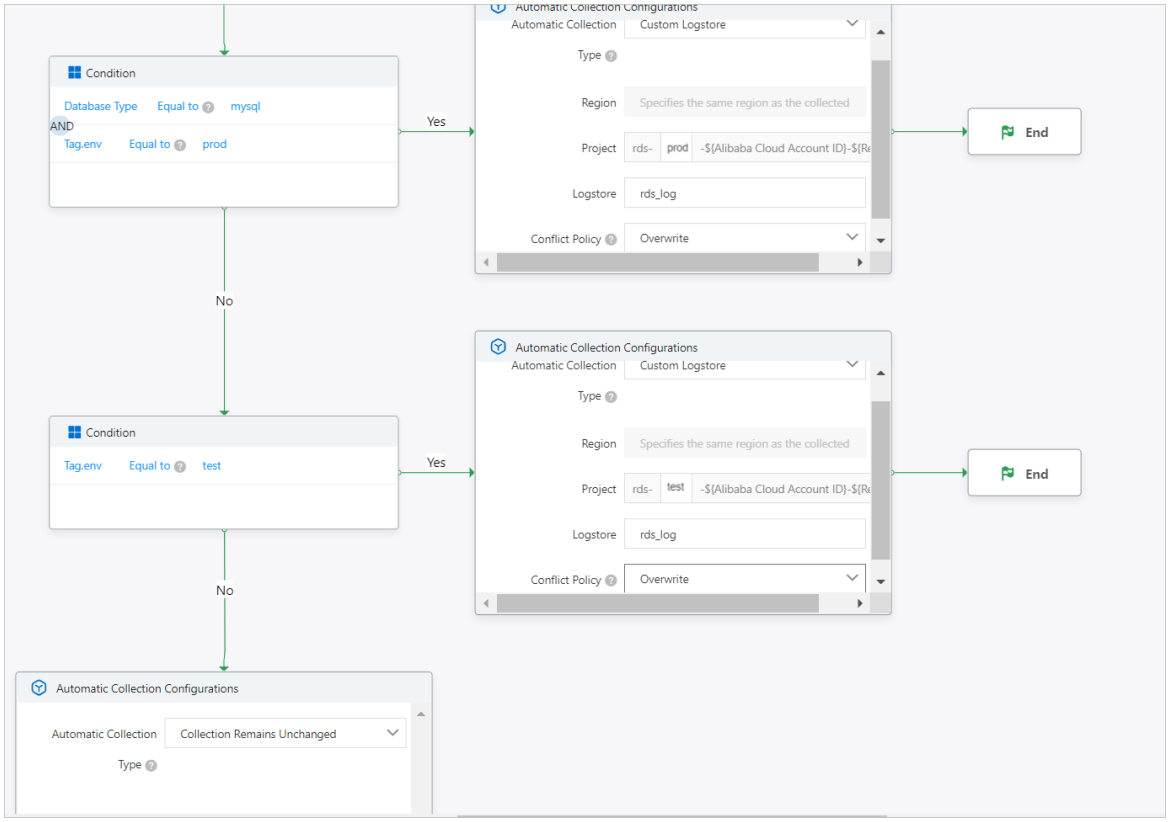
   In standard mode, multiple conditions are associated by the AND operator. In advanced mode, you can combine and nest conditions based on your business requirements.
6. Set parameters to configure automatic collection.

Parameter	Description
Automatic Collection Type	Select an automatic collection type. Valid values: ○ **Custom Logstore**: Log Service automatically collects the audit logs of the RDS instances that meet the specified conditions and saves the collected logs to the related destination Logstores. If a destination project or destination Logstore does not exist, Log Service automatically creates a project or Logstore. ○ **Collection Remains Unchanged**: If you select **Collection Remains Unchanged**, you do not need to set the **Region**, **Project**, **Logstore**, or **Conflict Policy** parameter. ■ If you have not enabled the log collection feature, the automatic collection feature is not automatically enabled for the RDS instances even if the RDS instances meet the specified conditions. ■ If you have enabled the log collection feature, the related destination Logstores remain unchanged even if the RDS instances meet the specified conditions.

Parameter	Description
Region	Log Service automatically selects regions based on the regions where the RDS instances that meet the specified conditions reside. You cannot modify this parameter.
Project	A project named `rds-xxx-${Alibaba Cloud account ID}-${region}` is automatically created in the regions where the RDS instances that meet the specified conditions reside. Example: rds-test-12345674523-cn-hangzhou.
Logstore	In the `rds-xxx-${Alibaba Cloud account ID}-${region}` project, a Logstore named `rds_log` is automatically created.
Conflict Policy	If the new destination Logstores are inconsistent with the destination Logstores that are in use, Log Service selects destination Logstores based on one of the following conditions: ○ **Ignore**: Audit logs are sent to the destination Logstores that are in use. ○ **Overwrite**: Audit logs are sent to the new destination Logstores.

The following configurations show an example about how to configure automatic collection:

- The audit logs of the ApsaraDB RDS for MySQL instances that have the `env==prod` tag are sent to the `rds_log` Logstore of the `rds-prod-${Alibaba Cloud account ID}-${region}` project.
- The audit logs of the ApsaraDB RDS for MySQL instances that have the `env==test` tag are sent to the `rds_log` Logstore of the `rds-test-${Alibaba Cloud account ID}-${region}` project.
- The audit logs of other RDS instances are sent to the destination Logstores that are in use.



7. Click the icon.
8. In the upper-right corner of the page, click **Save**.

Related operations

Operation	Description
Manage RDS instances	In the **RDS Instances** section of the **Data Import** tab, you can view all RDS instances that belong to your Alibaba Cloud account. You can also view the regions where the RDS instances reside and the collection statuses of the RDS instances.
Disable the log collection feature	In the **RDS Instances** section of the **Data Import** tab, find the RDS instance for which you want to disable the log collection feature. Then, click **Disable** in the Actions column.
Modify destination projects and destination Logstores	In the **RDS Instances** section of the **Data Import** tab, find the RDS instance for which you want to modify the destination project and destination Logstore. Then, click **Change** in the Actions column.
Manage destination projects and destination Logstores	In the **Destination Logstores** section of the **Data Import** tab, you can view the Logstores that are used to store RDS audit logs and modify the retention period of log data in destination Logstores.

What's next

After RDS audit logs are collected and sent to a destination Logstore, you can perform the following operations:

- On the **Search** tab, select the destination Logstore to query and analyze logs. For more information, see Query and analyze logs.
- On the **Audit Operations Center** tab, **Audit Security Center** tab, or **Audit Performance Center** tab, select the destination Logstore and view the related dashboard.

10.4. Configure alerts

CloudLens for RDS provides built-in alert rules. You can enable the alert instances of alert rules to monitor databases in CloudLens for RDS in real time. This topic describes how to configure alerts.

Prerequisites

The data access configuration is complete. For more information, see Enable the log collection feature.

Context

CloudLens for RDS provides the following built-in resources for alerting: alert rules, alert policy, action policy, user group, and alert template. Before you use the built-in resources, take note of the following items:

- You can specify the built-in alert policy in an alert rule.

  > **Note** The built-in alert rules that are provided by CloudLens for RDS are associated with the built-in alert policy. You cannot disassociate the built-in alert policy from the alert rules or associate other alert policies with the alert rules.

- You can specify the built-in action policy in the built-in alert policy.
- You can specify the built-in user group and built-in alert template in the built-in action policy.

You can use built-in resources or custom resources to configure alerts. This topic uses built-in resources as an example. For more information about how to use custom resources, see Log Audit Service.

Step 1: Create users

1.
2. In the **Log Application** section, click **CloudLens for RDS**.
3. In the left-side navigation pane, click **Alerts**.
4. On the **Alerts** tab, choose **Alert Management > User Management**.
5. Create users.

   For more information, see Create users and user groups.

Step 2: Add users to the built-in user group

1. On the **Alerts** tab, choose **Alert Management > User Group Management**.
2. In the User Groups list, find the built-in user group whose ID is **sls.app.audit.builtin** and click **Edit** in the Actions column.
3. In the **Edit User Group** dialog box, add the users that you create from the **Available Members** section to the **Selected Members** section. Then, click **OK**.

Step 3: Enable alert instances

1. On the **Alerts** tab, click **Alert Rules/Incidents**.

2. In the alert rule list, find the alert rule that you want to use and click **Enable** in the Actions column.

   After you enable an alert instance, Log Service monitors databases in CloudLens for RDS in real time. If you want to enable multiple alert instances, click **Add**.

   For more information about the parameters of an alert rule, see Security of RDS instances.

References

Operation	Description
Configure whitelists	You can configure whitelists for alert rules. This way, alerts are not triggered by specific users, instance IDs, or IP addresses. The whitelist configurations vary based on alert rules. For more information, see Security of RDS instances.
Disable alert instances	If you disable an alert instance, the status in the **Status** column of the alert instance changes to **Not Enabled**, and alerts are no longer triggered based on the alert instance. The configurations of the alert rule are not deleted. If you want to enable the alert instance again, you do not need to reconfigure the parameters of the alert rule.
Pause alert instances	If you pause an alert instance, alerts are not triggered based on the alert instance for a specified period of time.
Resume alert instances	You can resume paused alert instances.
Delete alert instances	If you delete an alert instance, the status in the **Status** column of the alert instance changes to **Not Created**. The configurations of the alert rule are deleted. The configurations include the Alibaba Cloud account that created the alert rule. If you want to enable the alert instance again, you must reconfigure the parameters of the alert rule.
Upgrade alert instances	If a major upgrade is released for alert rules or if additional configurations are required after alert rules are upgraded, you are prompted to upgrade alert rules. In most cases, Log Service automatically upgrades alert rules.
Initialize alert assets	If the assets generated during alert initialization are deleted by mistake or if the alert assets fail to be initialized for the first time, you can perform this operation to forcibly initialize the alert assets.

10.5. Log fields

This topic describes the fields in SQL audit logs collected from ApsaraDB RDS instances.

Log field	Description
__topic__	The topic of a log. The value is fixed as rds_audit_log.

Log field	Description
instance_id	The ID of an RDS instance.
check_rows	The number of scanned rows.
db	The name of a database.
fail	Indicates whether an SQL statement is successfully executed. • 0: successful • 1: failed
client_ip	The IP address of a client that accesses an RDS instance.
latency	The time required to return the results of an SQL statement. Unit: microseconds.
origin_time	The point in time at which an SQL statement is executed.
return_rows	The number of returned rows.
sql	The SQL statement that is executed.
thread_id	The ID of a thread.
user	The username of a user who executes an SQL statement.
update_rows	The number of updated rows.

11.CloudLens for ALB (formerly ALB Log Center)

11.1. Usage notes

Log Service and Server Load Balancer (SLB) jointly launch the CloudLens for ALB application. You can use the application to analyze the Layer 7 access logs of Application Load Balancer (ALB), analyze the metrics that are aggregated at one-second intervals, and generate alerts in real time. The application also provides AIOps-based automated anomaly detection. The application allows you to analyze the behavior, geographical distribution, request success rates, and response latency of clients. This topic describes the features, assets, billing, and limits of the CloudLens for ALB application.

Features

The CloudLens for ALB application automatically aggregates real-time access logs and provides features such as intelligent inspection and real-time alerting.

- The application allows you to manage all ALB instances within your Alibaba Cloud account in a centralized manner.
- The application allows you to enable the data collection feature for the access logs of ALB instances with a few clicks and manage the collection status of the instances in a centralized manner.
- The application allows you to store, query, and analyze ALB access logs in real time.
- The application extracts various metrics in real time based on raw access logs. The metrics include page views (PVs), request success rates, average latency, P50 latency, P99 latency, P9999 latency, and inbound and outbound traffic. The application can extract metrics from one or more of the following dimensions: app_lb_id, host, and status.
- The application provides various reports, including Monitoring Center, Real-time Monitoring, and Instance Inspection. You can subscribe to the reports by using emails or the webhooks of DingTalk groups.
- The application provides the intelligent inspection feature and supports global inspection and app_lb_id-based inspection. You can label anomalies in reports.
- The application supports custom alert settings and can send alert notifications by using the following methods: Message Center, text messages, emails, voice calls, DingTalk, and custom webhooks.

Benefits

- Easy-to-use: You can enable the application with a few clicks and use centralized storage for the application. You do not need to focus on the collection, storage, computing, or visualization of logs. This allows developers and O&M personnel to focus on business development and technical research without the need to worry about tedious and time-consuming log processing.
- Capable of processing large amounts of data: The number of ALB access logs increases with the number of PVs for ALB instances. As a result, a large number of access logs are accumulated. When you process the large number of access logs, you must balance performance and costs.
- Real-time: Real-time data is required in scenarios such as DevOps, monitoring, and alerting. The application integrates Alibaba Cloud SLB with the big data computing capabilities of Log Service. This allows the application to analyze and process real-time logs in seconds.
- Flexible: You can enable or disable the data collection feature for each of your ALB instances. You

can also specify a custom retention period for logs. The storage capacity of a Logstore can be dynamically scaled to meet service requirements.

- Intelligent: The application automatically inspects ALB metrics to identify and locate errors in an efficient and accurate manner. The inspection is based on the AIOps algorithms that are developed by Alibaba DAMO Academy.

Assets

You can view the assets of the CloudLens for ALB application in the project that you specify when you enable the data collection feature. The following assets are included:

- Logstores
  - A Logstore that is used to store the Layer 7 access logs of ALB instances. You can create the Logstore.
  - A Logstore that is used to store inspection results. After you enable the data collection feature, Log Service automatically generates a dedicated Logstore named *The name of the Logstore for access logs*-metrics-result.

> **Notice**
> - Do not delete the Logstore that is used to store the Layer 7 access logs of ALB instances. If you delete the Logstore, access logs cannot be collected or sent to Log Service.
> - Do not delete the indexes of specific fields in the Logstore that is used to store access logs. If you delete the indexes, metric conversion fails.

- Metricstore

  A Metricstore that is used to store aggregated data about the collected metrics. After you enable the data collection feature, Log Service automatically generates a dedicated Metricstore named *The name of the Logstore for access logs*-metrics.

> **Note** The Metricstore stores aggregated data about the collected metrics. The data volume significantly reduces after aggregation, and you can store the aggregated data for a long period of time.

- Aggregation rules

Rule name	Time granularity	Dimension	New metric

Rule name	Time granularity	Dimension	New metric
total	10 seconds	total	◦ pv ◦ body_bytes_sent_avg ◦ body_bytes_sent_sum ◦ request_length_avg ◦ request_length_sum ◦ upstream_response_time_avg ◦ upstream_response_time_p50 ◦ upstream_response_time_p90 ◦ upstream_response_time_p99 ◦ request_time_avg ◦ request_time_p50 ◦ request_time_p90 ◦ request_time_p99 ◦ request_time_p9999
app_lb_id	10 seconds	app_lb_id	◦ pv:app_lb_id ◦ body_bytes_sent_avg:app_lb_id ◦ body_bytes_sent_sum:app_lb_id ◦ request_length_avg:app_lb_id ◦ request_length_sum:app_lb_id ◦ upstream_response_time_avg:app_lb_id ◦ upstream_response_time_p50:app_lb_id ◦ upstream_response_time_p90:app_lb_id ◦ upstream_response_time_p99:app_lb_id ◦ request_time_avg:app_lb_id ◦ request_time_p50:app_lb_id ◦ request_time_p90:app_lb_id ◦ request_time_p99:app_lb_id ◦ request_time_p9999:app_lb_id

Rule name	Time granularity	Dimension	New metric
app_lb_id_host_status	10 seconds	app_lb_id+host+status	◦ pv:app_lb_id:host:status ◦ body_bytes_sent_avg:app_lb_id:host:status ◦ body_bytes_sent_sum:app_lb_id:host:status ◦ request_length_avg:app_lb_id:host:status ◦ request_length_sum:app_lb_id:host:status ◦ upstream_response_time_avg:app_lb_id:host:status ◦ upstream_response_time_p50:app_lb_id:host:status ◦ upstream_response_time_p90:app_lb_id:host:status ◦ upstream_response_time_p99:app_lb_id:host:status ◦ request_time_avg:app_lb_id:host:status ◦ request_time_p50:app_lb_id:host:status ◦ request_time_p90:app_lb_id:host:status ◦ request_time_p99:app_lb_id:host:status ◦ request_time_p9999:app_lb_id:host:status

- Inspection rules

Rule name	Algorithm	Metric

Rule name	Algorithm	Metric
alb-patrol-total	Time2Graph	○ pv ○ body_bytes_sent_avg ○ body_bytes_sent_sum ○ request_length_avg ○ request_length_sum ○ upstream_response_time_avg ○ upstream_response_time_p50 ○ upstream_response_time_p90 ○ upstream_response_time_p99 ○ request_time_avg ○ request_time_p50 ○ request_time_p90 ○ request_time_p99 ○ request_time_p9999
alb-patrol-alb	Time2Graph	○ pv:alb ○ body_bytes_sent_avg:alb ○ body_bytes_sent_sum:alb ○ request_length_avg:alb ○ request_length_sum:alb ○ upstream_response_time_avg:alb ○ upstream_response_time_p50:alb ○ upstream_response_time_p90:alb ○ upstream_response_time_p99:alb ○ request_time_avg:alb ○ request_time_p50:alb ○ request_time_p90:alb ○ request_time_p99:alb ○ request_time_p9999:alb

- Dedicated dashboards

Dashboard	Associated Logstore or Metricstore	Description
Monitoring Overview	*The name of the Logstore for access logs*-metrics	Displays the overall information about the metrics of an ALB instance. The metrics include Core Indicators, Error Code, Traffic, Exception Event, PVs, Access Success Rate, Traffic, and Avg Latency.

Dashboard	Associated Logstore or Metricstore	Description
Monitoring Center	◦ *The name of the Log store for access logs*-metrics ◦ *The name of the Log store for access logs*	Displays the real-time monitoring data of an ALB instance. The data includes PVs, Request Success Rate, Average Latency, Requests with Status Code 4xx, Status Distribution, Traffic, P50 Latency, P90 Latency, P99 Latency, P9999 Latency, Hosts with Most Requests, Hosts with Highest Latency, Hosts with Highest Failure Rate, URLs with Most Requests, URLs with Highest Latency, URLs with Highest Failure Rate, Backends with Most Requests, Backends with Highest Latency, and Backends with Highest Failure Rate.
Real-time Monitoring	*The name of the Logstore for access logs*	Displays the metrics that are aggregated at one-second intervals. You can use the metrics to identify exceptions that are related to transient jitters. The metrics include QPS, Access Latency, Upstream Latency, Success Rate, Request Traffic, Response Body Traffic, Status Code 2xx, Status Code 3xx, Error Codes, Upstream Status Code 2xx, Upstream Status Code 3xx, and Upstream Error Codes.
Instance Inspection	◦ *The name of the Log store for access logs*-metrics ◦ *The name of the Log store for access logs*-metrics-result	Displays the information about anomalies that are detected by Log Service in an ALB instance. The detection is based on the machine learning algorithms that are provided by Log Service. The information includes Exceptions, High Exceptions, Exception Distributions, Middle Exceptions, Low Exceptions, Exception Distributions, Exception List, and Exception Events.
Access Overview	*The name of the Logstore for access logs*	Displays the status of an ALB instance. The status information includes PVs (Day-on-day), PVs (Week-on-week), UVs (Day-on-day), UVs (Week-on-week), PV Distribution, UV Distribution, PVs Today, PVs of 7 Days, Top 10 States with Most Requests, Percentage of Mobile Users, TOP 10 Hosts with Most Requests, TOP 10 User Agents with Most Requests, and IP Addresses with Most Requests.

Billing

- You are not charged for the log management feature of SLB.
- After ALB access logs are collected and sent to Log Service, you are charged for billable items such as the storage space, read traffic, number of requests, data transformation, and data shipping. The fees are included in your Log Service bills. For more information, see Billable items.

Limits

- The project that you specify in Log Service must reside in the same region as the ALB instance that you use.

- CloudLens for ALB is supported in the following regions.

Cloud type	Region
Alibaba Cloud public cloud	China (Qingdao), China (Beijing), China (Zhangjiakou), China (Ulanqab), China (Hangzhou), China (Shanghai), China (Nanjing - Local Region), China (Shenzhen), China (Guangzhou), China (Chengdu), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Japan (Tokyo), India (Mumbai), US (Silicon Valley), US (Virginia), and Germany (Frankfurt)

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

> **Notice** If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

11.2. Grant the operation permissions on CloudLens for ALB to a RAM user

This topic describes how to grant the operation permissions on CloudLens for ALB to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant the operation permissions on CloudLens for ALB to a RAM user in one of the following modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode allows you to perform fine-grained access control. However, this mode requires complex configurations.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the code editor with one of the following scripts, and then click **Next: Edit Basic Information**.

   You can grant the read-only permissions or read and write permissions on CloudLens for ALB to the RAM user.

   - Read-only permissions: Use the following script to authorize the RAM user only to view each page of CloudLens for ALB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "log:GetProductDataCollection",
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:alb:*:*:loadbalancer/*"
            ],
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

   - Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by CloudLens for ALB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
```

```
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:GetProductDataCollection",
                "log:OpenProductDataCollection",
                "log:CloseProductDataCollection"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:alb:*:*:loadbalancer/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.alb.global_conf.standard_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
```

```
                    "ram:ServiceName": "audit.log.aliyuncs.com"
                }
            }
        ],
        "Version": "1"
}
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the policy name to log-alb-policy.

3. Attach the policy to the RAM user.
   i. In the left-side navigation pane, choose **Identities > Users**.
   ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.
   iii. In the **Add Permissions** panel, go to the **Select Policy** section, click **Custom Policy**, and then click the policy that you create in Step . In this example, click log-alb-policy.
   iv. Click **OK**.

11.3. Enable data collection

CloudLens for ALB allows you to enable data collection for the access logs of Application Load Balancer (ALB) instances with a few clicks. This topic describes how to enable data collection in CloudLens for ALB. This topic also describes the operations that you can perform after you enable data collection.

Prerequisites

- An ALB instance is created. For more information, see Create an ALB instance.
- A project and a Logstore are created in Log Service. For more information, see Create a project and Create a Logstore.

Authorization

Notice You need to perform this operation only once.

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click **CloudLens for ALB**.
3. Follow the on-screen instructions to enable CloudLens for ALB.

   When you enable the application, Log Service automatically authorizes CloudLens for ALB to assume the AliyunServiceRolePolicyForSLSAudit service-linked role to collect ALB access logs. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

Enable data collection

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click **CloudLens for ALB**.
3. On the **ALB Instance Access** tab of the **Access Management** page, find the ALB instance for which you want to enable data collection and click **Enable**.
4. In the **Enable Access Logs Collect** dialog box, select the project and the Logstore. Then, click

Confirm.

What to do next

After you enable CloudLens for ALB, you can perform the following operations on the **Access Management** page.

Operation	Description
Manage ALB instances	After you enable CloudLens for ALB, CloudLens for ALB displays all ALB instances within your Alibaba Cloud account. Click the ALB instance that you want to manage. Then, you are navigated to the SLB console. You can view the details about the ALB instance and create a listener for the instance. For more information, see View details of an ALB instance.
Disable data collection	Find the ALB instance for which you want to disable data collection and click **Disable** in the Access Logs column.
Query and analyze data	Find the ALB instance whose data you want to query and analyze and click **Access Logs** in the Actions column. Then, you are navigated to the Logstore in which the data is stored. You can view, query, and analyze the raw logs. For more information, see Query and analyze logs.
Modify data retention periods	On the **Destination Logstore** tab, find the Logstore that you want to manage and click the icon to modify the retention period of data in the Logstore.
Use the updated data collection feature	If you enable log shipping for your ALB instance in the SLB console and enable log collection in the Log Service console by using CloudLens for ALB, CloudLens for ALB automatically collects access logs, but does not automatically detect anomalies or extracts metrics. You can click **Upgrade**. Then, CloudLens for ALB automatically detects anomalies, extracts metrics, and generates assets such as detection results, Metricstores, and inspection jobs.

What's next

View reports

11.4. View reports

CloudLens for ALB provides multiple dashboards to display the information about an Application Load Balancer (ALB) instance in different dimensions. The dashboards are Monitoring Overview, Monitoring Center, Real-time Monitoring, Instance Inspection, and Access Overview.

Prerequisites

An ALB instance is created, and log collection is enabled for the instance. For more information, see Enable data collection.

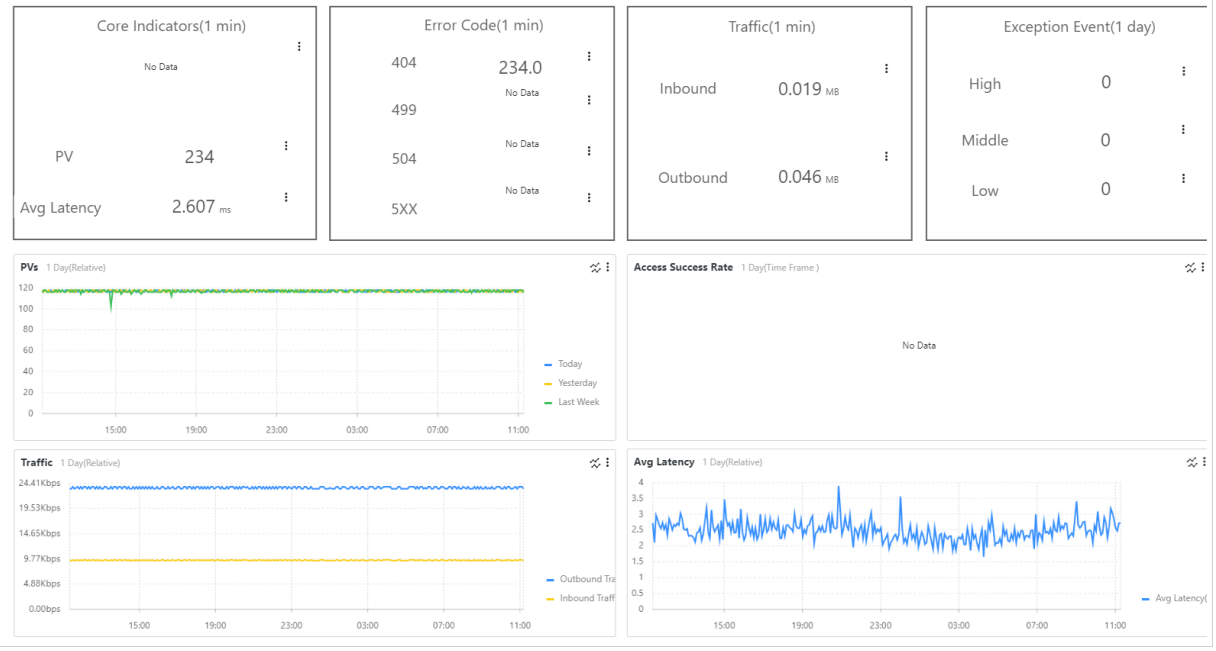
Entry point

1.

2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click **CloudLens for ALB**.
3. In the left-side navigation pane, click **Report Center**.
4. In the upper-left corner of the page that appears, select the ALB instance.

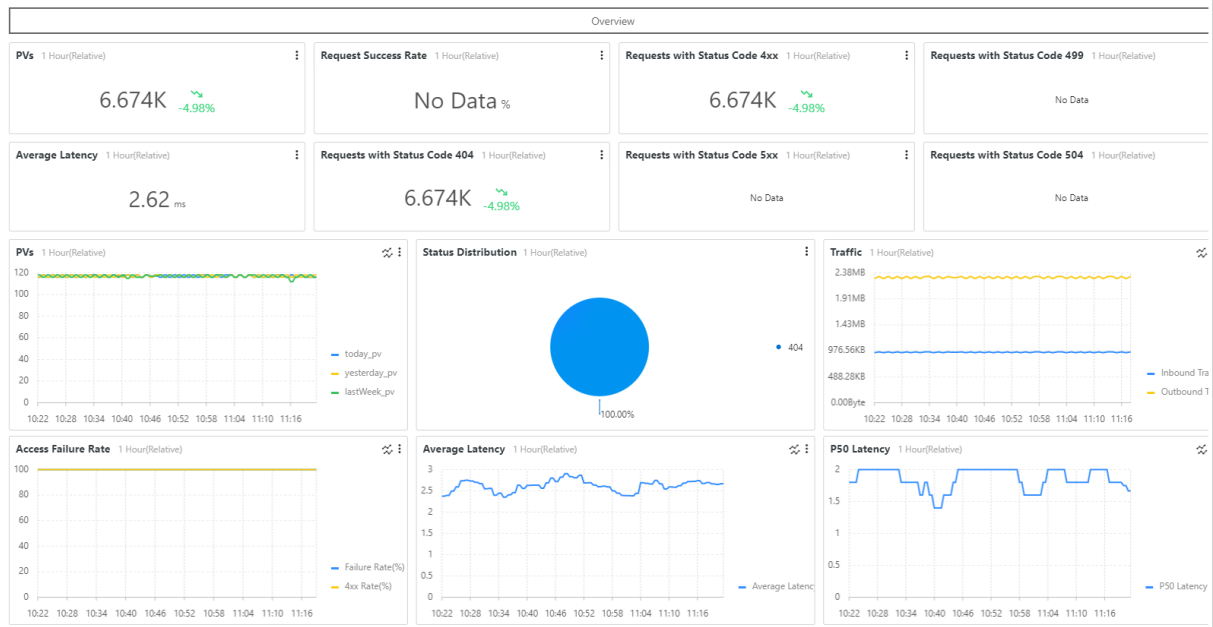
Monitoring Overview

The **Monitoring Overview** dashboard displays the overall information about the metrics of an ALB instance. The metrics include Core Indicators, Error Code, Traffic, Exception Event, PVs, Access Success Rate, Traffic, and Avg Latency.



Monitoring Center

The **Monitoring Center** dashboard displays the real-time monitoring data of an ALB instance. The data includes PVs, Request Success Rate, Average Latency, Requests with Status Code 4xx, Status Distribution, Traffic, P50 Latency, P90 Latency, P99 Latency, P9999 Latency, Hosts with Most Requests, Hosts with Highest Latency, Hosts with Highest Failure Rate, URLs with Most Requests, URLs with Highest Latency, URLs with Highest Failure Rate, Backends with Most Requests, Backends with Highest Latency, and Backends with Highest Failure Rate.

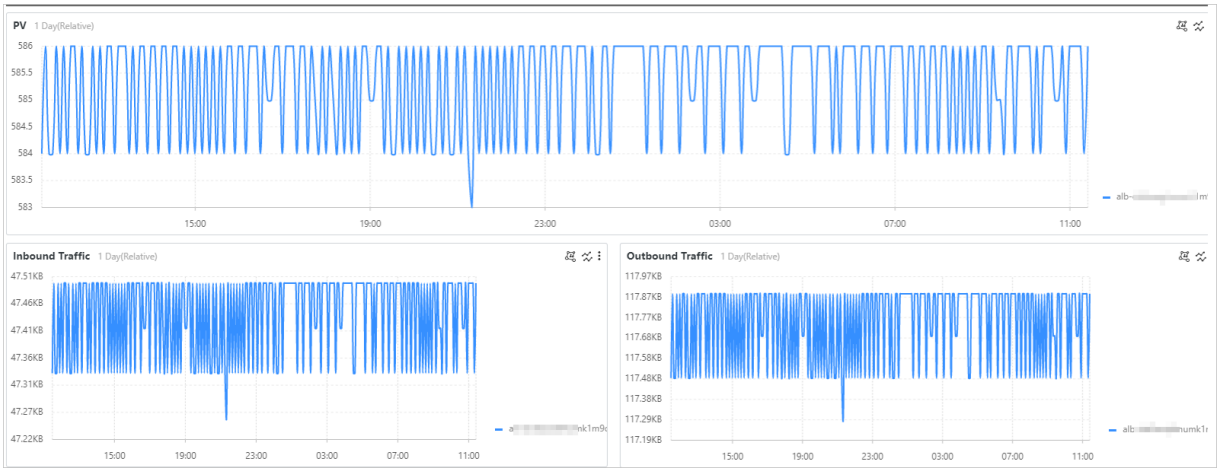


Real-time Monitoring

The **Real-time Monitoring** dashboard displays the metrics of an ALB instance that are aggregated at one-second intervals. The metrics include QPS, Access Latency, Upstream Latency, Success Rate, Request Traffic, Response Body Traffic, Status Code 2xx, Status Code 3xx, Error Codes, Upstream Status Code 2xx, Upstream Status Code 3xx, and Upstream Error Codes.

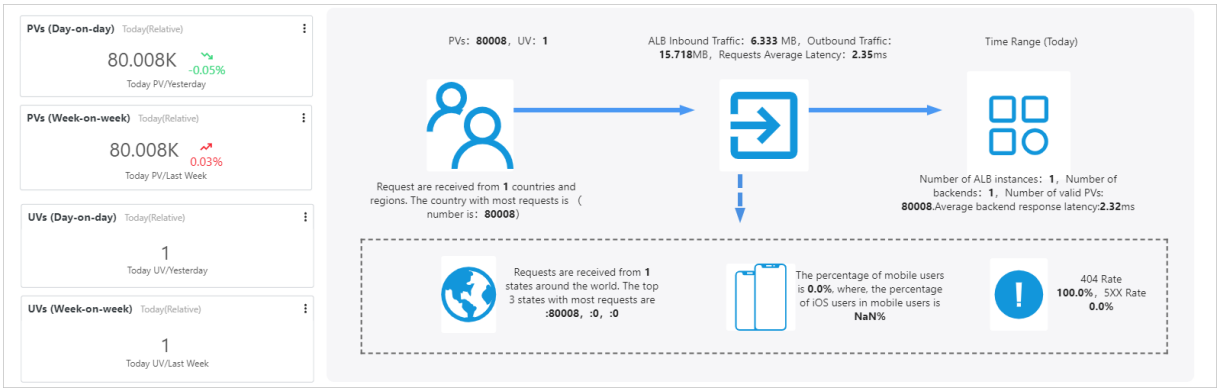
Instance Inspection

The **Instance Inspection** dashboard displays the information about anomalies that are detected by Log Service in an ALB instance. The detection is based on the machine learning algorithms that are provided by Log Service. The information includes Exceptions, High Exceptions, Exception Distributions, Middle Exceptions, Low Exceptions, Exception Distributions, Exception List, and Exception Events.



Access Overview

The **Access Overview** dashboard displays the status of an ALB instance. The status information includes PVs (Day-on-day), PVs (Week-on-week), UVs (Day-on-day), UVs (Week-on-week), PV Distribution, UV Distribution, PVs Today, PVs of 7 Days, Top 10 States with Most Requests, Percentage of Mobile Users, TOP 10 Hosts with Most Requests, TOP 10 User Agents with Most Requests, and IP Addresses with Most Requests.



11.5. Metrics

This topic describes the metrics that are extracted from the Layer 7 access logs of Application Load Balancer (ALB). The metrics include global metrics and the metrics that are specific to app_lb_id, status, and upstream_status.

The metrics in this topic use the time series data format. You can execute PromQL or SQL statements to query and analyze the metrics. For more information, see Overview of query and analysis on time series data.

Global metrics

The following table describes the global metrics.

Metric	Description
pv	The total number of visits.
body_bytes_sent_avg	The average number of bytes in the bodies of the responses that are sent to clients.
body_bytes_sent_sum	The total number of bytes in the bodies of the responses that are sent to clients.
request_length_avg	The average length of requests.
request_length_sum	The total length of requests.
request_time_avg	The average period of time that is consumed to respond to a request.
request_time_p50	The period of time that is consumed to respond to a request at the 50th percentile among all periods of time.
request_time_p90	The period of time that is consumed to respond to a request at the 90th percentile among all periods of time.
request_time_p99	The period of time that is consumed to respond to a request at the 99th percentile among all periods of time.
request_time_p9999	The period of time that is consumed to respond to a request at the 99.99th percentile among all periods of time.
upstream_response_time_avg	The average period of time for a request. **Note** The upstream_response_time metric indicates the interval between the time when an ALB instance is connected to the backend server and the time when the ALB instance is disconnected from the backend server after the required data is received.
upstream_response_time_p50	The period of time at the 50th percentile among the periods of time for all requests.
upstream_response_time_p90	The period of time at the 90th percentile among the periods of time for all requests.
upstream_response_time_p99	The period of time at the 99th percentile among the periods of time for all requests.

Metrics specific to app_lb_id

The tag of the metrics specific to app_lb_id is app_lb_id. The following table describes the metrics.

Metric	Description
pv:app_lb_id	The number of visits to an ALB instance.

Metric	Description
body_bytes_sent_avg:app_lb_id	The average number of bytes in the bodies of the responses that are sent to clients.
body_bytes_sent_sum:app_lb_id	The total number of bytes in the bodies of the responses that are sent to clients.
request_length_avg:app_lb_id	The average length of requests.
request_length_sum:app_lb_id	The total length of requests.
request_time_avg:app_lb_id	The average period of time that is consumed to respond to a request.
request_time_p50:app_lb_id	The period of time that is consumed to respond to a request at the 50th percentile among all periods of time.
request_time_p90:app_lb_id	The period of time that is consumed to respond to a request at the 90th percentile among all periods of time.
request_time_p99:app_lb_id	The period of time that is consumed to respond to a request at the 99th percentile among all periods of time.
request_time_p9999:app_lb_id	The period of time that is consumed to respond to a request at the 99.99th percentile among all periods of time.
upstream_response_time_avg:app_lb_id	The average period of time for a request. **Note** The upstream_response_time metric indicates the interval between the time when an ALB instance is connected to the backend server and the time when the ALB instance is disconnected from the backend server after the required data is received.
upstream_response_time_p50:app_lb_id	The period of time at the 50th percentile among the periods of time for all requests.
upstream_response_time_p90:app_lb_id	The period of time at the 90th percentile among the periods of time for all requests.
upstream_response_time_p99:app_lb_id	The period of time at the 99th percentile among the periods of time for all requests.

Metrics specific to status

The tag of the metrics specific to status is app_lb_id+host+status. The following table describes the metrics.

Metric	Description
pv:app_lb_id:host:status	The number of visits in the dimension of status.

Metric	Description
body_bytes_sent_avg:app_lb_id:host:status	The average number of bytes in the bodies of the responses that are sent to clients.
body_bytes_sent_sum:app_lb_id:host:status	The total number of bytes in the bodies of the responses that are sent to clients.
request_length_avg:app_lb_id:host:status	The average length of requests.
request_length_sum:app_lb_id:host:status	The total length of requests.
request_time_avg:app_lb_id:host:status	The average period of time that is consumed to respond to a request.
request_time_p50:app_lb_id:host:status	The period of time that is consumed to respond to a request at the 50th percentile among all periods of time.
request_time_p90:app_lb_id:host:status	The period of time that is consumed to respond to a request at the 90th percentile among all periods of time.
request_time_p99:app_lb_id:host:status	The period of time that is consumed to respond to a request at the 99th percentile among all periods of time.
request_time_p9999:app_lb_id:host:status	The period of time that is consumed to respond to a request at the 99.99th percentile among all periods of time.
upstream_response_time_avg:app_lb_id:host:status	The average period of time for a request. **Note** The upstream_response_time metric indicates the interval between the time when an ALB instance is connected to the backend server and the time when the ALB instance is disconnected from the backend server after the required data is received.
upstream_response_time_p50:app_lb_id:host:status	The period of time at the 50th percentile among the periods of time for all requests.
upstream_response_time_p90:app_lb_id:host:status	The period of time at the 90th percentile among the periods of time for all requests.
upstream_response_time_p99:app_lb_id:host:status	The period of time at the 99th percentile among the periods of time for all requests.

11.6. Log fields

This topic describes the fields in Layer 7 access logs of Application Load Balancer (ALB).

Log field	Description
__topic__	The topic of a log. Valid value: alb_layer7_access_log.
body_bytes_sent	The number of bytes in the body of an HTTP response that is sent to a client.
client_ip	The IP address of a client.
host	The domain name or IP address of a server. By default, the value is obtained from request parameters. If no value is obtained from request parameters, the value is obtained from the host header field. If no value is obtained from request parameters or the host header field, the IP address of the backend server that processes the request is used as the value.
http_host	The host header field of a request.
http_referer	The URL of a source web page.
http_user_agent	The browser information of a client.
http_x_forwarded_for	The client IP address that is recorded after a request of a client is forwarded by the proxy.
http_x_real_ip	The real IP address of a client.
read_request_time	The duration in which the proxy reads a request message. Unit: milliseconds.
request_length	The length of a request. The request line, request headers, and request body are all counted.
request_method	The request method.
request_time	The interval between the time when the proxy receives the first request and the time when the proxy returns a response. Unit: seconds.
request_uri	The URI of a request that is received by the proxy.
scheme	The schema of a request. Valid values: HTTP and HTTPS.
server_protocol	The HTTP version of a request that is received by the proxy. Example: HTTP/1.0 or HTTP/1.1.
slb_vport	The listening port of a Server Load Balancer (SLB) instance.
app_lb_id	The ID of an ALB instance.
ssl_cipher	The cipher suite that is used to establish an SSL connection. Example: ECDHE-RSA-AES128-GCM-SHA256.
ssl_protocol	The protocol that is used to establish an SSL connection. Example: TLSv1.2.

Log field	Description
status	The HTTP status code that is sent by the proxy.
tcpinfo_rtt	The round-trip time (RTT) of a client TCP connection. Unit: microseconds.
time	The time when the log is generated.
upstream_addr	The IP address and port number of the backend server.
upstream_response_time	The interval between the time when an ALB instance is connected to the backend server and the time when the ALB instance is disconnected from the backend server after the required data is received.
upstream_status	The HTTP status code that is received by the proxy from the backend server.
vip_addr	The virtual IP address.
write_response_time	The period of time that is consumed by the proxy to respond to a write request. Unit: milliseconds.

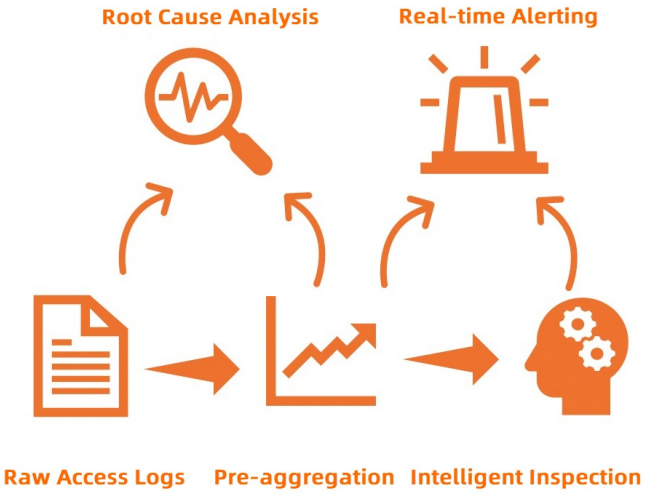
12.CloudLens for CLB (formerly SLB Log Center)

12.1. Usage notes

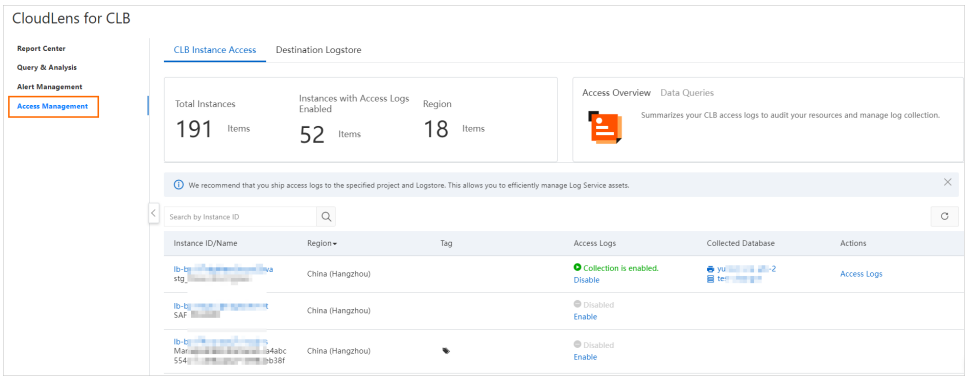
Log Service and Alibaba Cloud Server Load Balancer (SLB) jointly launch the CloudLens for CLB application. You can use the application to analyze Classic Load Balancer (CLB) access logs, analyze metrics in seconds, and generate alerts in real time. The application also provides AIOps-based automated anomaly detection. The CloudLens for CLB application allows you to analyze the behavior, geographical distribution, request success rate, and response latency of clients. This topic describes the features, benefits, assets, billing, and limits of the CloudLens for CLB application.

Features

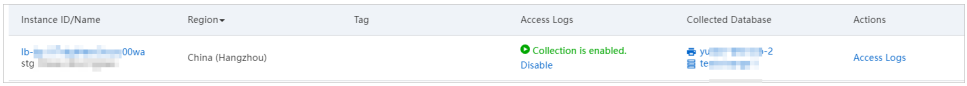
The CloudLens for CLB application automatically aggregates real-time access logs and provides features such as intelligent inspection and real-time alerting.



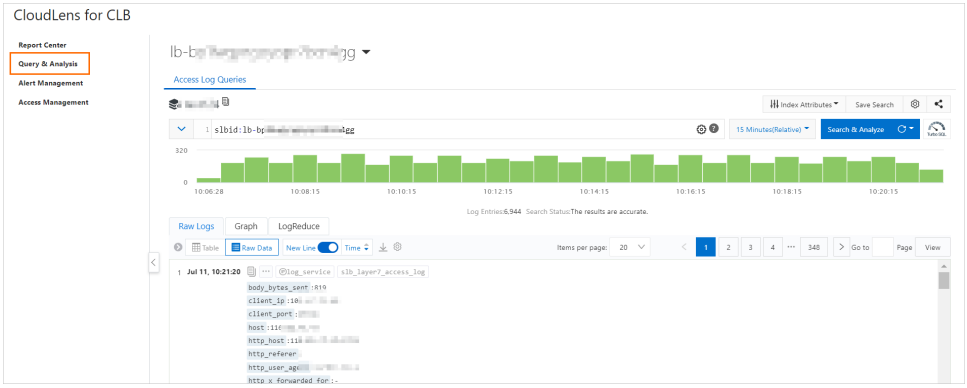
- The application allows you to manage all CLB instances within your Alibaba Cloud account in a centralized manner.



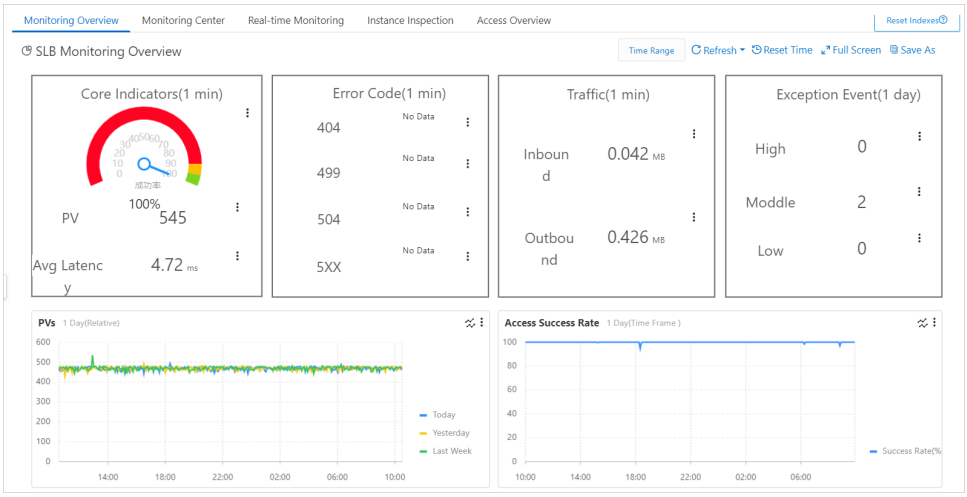
- The application allows you to enable the log collection feature for CLB access logs with a few clicks and manage the log collection status of the CLB instances in a centralized manner.



- The application allows you to store, query, and analyze CLB access logs in real time.



- The application provides various reports, including Monitoring Center and Instance Inspection. You can subscribe to these reports and configure settings to send the reports by using emails or DingTalk group messages.



- The application supports custom alert settings and can send alert notifications by using the following methods: Message Center, text messages, emails, voice calls, DingTalk, and custom webhooks.
- The application extracts various metrics in real time based on raw access logs. The metrics include page views (PVs), request success rates, average latency, P50 latency, P99 latency, and P9999 latency, and inbound and outbound traffic. The application can extract metrics from one or more of the following dimensions: slbid, host, method, and status.
- The application provides the intelligent inspection feature and supports global inspection and slbid-based inspection. You can label anomalies in reports.

Benefits

- Easy-to-use: You can enable the application with a few clicks and use centralized storage for the application. You do not need to focus on the collection, storage, computing, and visualization of logs. This allows developers and O&M personnel to focus on business development and technical research without the need to worry about tedious and time-consuming log processing.
- Capable of processing large amounts of data: The number of CLB access logs increases with the number of PVs for CLB instances. This way, a large number of access logs are accumulated. When you

process the large number of access logs, you must balance performance and costs. In this case, you can use the CloudLens for CLB application, which can improve query performance. The application supports custom settings for the pre-aggregation feature and can use the feature to calculate and aggregate metrics in real time. After the aggregation, the data volume significantly reduces.

- Real-time: Real-time data is required in scenarios such as DevOps, monitoring, and alerting. The application can analyze and process real-time logs in seconds by integrating Alibaba Cloud SLB with the big data computing capabilities of Log Service.
- Flexible: You can enable or disable the access log management feature based on the specifications of your CLB instances. You can also specify a custom retention period for logs. The storage capacity of a Logstore can be dynamically scaled to meet service requirements.
- Intelligent: The application automatically inspects CLB metrics to identify and locate errors in an efficient and accurate manner based on the AIOps algorithms that are developed by Alibaba DAMO Academy.

Assets

You can view the assets of the CloudLens for CLB application in the project that you specify. The following assets are included:

- Logstores
  - The Logstore that is used to store CLB access logs. You can use a custom Logstore.
  - The Logstore that is used to store inspection results. After you enable the log collection feature for access logs, Log Service automatically generates a dedicated Logstore named *Logstore for access logs*-metrics-result.

> **Note**
> - Do not delete the Logstore that is used to store CLB access logs. If you delete the Logstore, access logs cannot be collected to Log Service.
> - Do not delete the indexes of specific fields in the Logstore that is used to store CLB access logs. If you delete the indexes, metric conversion fails.

- Metricstore

  The Metricstore is used to store the metrics that are generated after raw metrics are aggregated. After you enable the log collection feature for access logs, Log Service automatically generates a dedicated Metricstore named *Logstore for access logs*-metrics.

> **Note** The Metricstore stores the metrics that are generated after raw metrics are aggregated. The data volume significantly reduces after aggregation, and you can store aggregated metrics for a long period of time.

- Aggregation rules

Rule name	Time granularity	Dimension	New metric

Rule name	Time granularity	Dimension	New metric
total	10 seconds	total	◦ pv ◦ body_bytes_sent_avg ◦ body_bytes_sent_sum ◦ request_length_avg ◦ request_length_sum ◦ upstream_response_time_avg ◦ upstream_response_time_p50 ◦ upstream_response_time_p90 ◦ upstream_response_time_p99 ◦ upstream_response_time_p9999 ◦ request_time_avg ◦ request_time_p50 ◦ request_time_p90 ◦ request_time_p99 ◦ request_time_p9999
slbid	10 seconds	slbid	◦ pv:slb ◦ body_bytes_sent_avg:slb ◦ body_bytes_sent_sum:slb ◦ request_length_avg:slb ◦ request_length_sum:slb ◦ upstream_response_time_avg:slb ◦ upstream_response_time_p50:slb ◦ upstream_response_time_p90:slb ◦ upstream_response_time_p99:slb ◦ upstream_response_time_p9999:slb ◦ request_time_avg:slb ◦ request_time_p50:slb ◦ request_time_p90:slb ◦ request_time_p99:slb ◦ request_time_p9999:slb

Rule name	Time granularity	Dimension	New metric
slbid_host_status	10 seconds	slbid+host+status	◦ pv:slbid:host:status ◦ body_bytes_sent_avg:slbid:host:status ◦ body_bytes_sent_sum:slbid:host:status ◦ request_length_avg:slbid:host:status ◦ request_length_sum:slbid:host:status ◦ upstream_response_time_avg:slbid:host:status ◦ upstream_response_time_p50:slbid:host:status ◦ upstream_response_time_p90:slbid:host:status ◦ upstream_response_time_p99:slbid:host:status ◦ upstream_response_time_p9999:slbid:host:status ◦ request_time_avg:slbid:host:status ◦ request_time_p50:slbid:host:status ◦ request_time_p90:slbid:host:status ◦ request_time_p99:slbid:host:status ◦ request_time_p9999:slbid:host:status

Rule name	Time granularity	Dimension	New metric
slbid+host+status+request_method+upstream_status+url	10 seconds	slbid+host+status+request_method+upstream_status+url	◦ pv:slbid:host:status:method:upstream_status ◦ body_bytes_sent_avg:slbid:host:status:method:upstream_status ◦ body_bytes_sent_sum:slbid:host:status:method:upstream_status ◦ request_length_avg:slbid:host:status:method:upstream_status ◦ request_length_sum:slbid:host:status:method:upstream_status ◦ upstream_response_time_avg:slbid:host:status:method:upstream_status ◦ upstream_response_time_p50:slbid:host:status:method:upstream_status ◦ upstream_response_time_p90:slbid:host:status:method:upstream_status ◦ upstream_response_time_p99:slbid:host:status:method:upstream_status ◦ upstream_response_time_p9999:slbid:host:status:method:upstream_status ◦ request_time_avg:slbid:host:status:method:upstream_status ◦ request_time_p50:slbid:host:status:method:upstream_status ◦ request_time_p90:slbid:host:status:method:upstream_status ◦ request_time_p99:slbid:host:status:method:upstream_status ◦ request_time_p9999:slbid:host:status:method:upstream_status

- Inspection rules

Rule name	Status	Algorithm	Metric

Rule name	Status	Algorithm	Metric
slb-patrol-total	This rule is enabled by default.	Time2Graph	◦ pv ◦ body_bytes_sent_avg ◦ body_bytes_sent_sum ◦ request_length_avg ◦ request_length_sum ◦ upstream_response_time_avg ◦ upstream_response_time_p50 ◦ upstream_response_time_p90 ◦ upstream_response_time_p99 ◦ upstream_response_time_p9999 ◦ request_time_avg ◦ request_time_p50 ◦ request_time_p90 ◦ request_time_p99 ◦ request_time_p9999
slb-patrol-slb	This rule is enabled by default.	Time2Graph	◦ pv:slb ◦ body_bytes_sent_avg:slb ◦ body_bytes_sent_sum:slb ◦ request_length_avg:slb ◦ request_length_sum:slb ◦ upstream_response_time_avg:slb ◦ upstream_response_time_p50:slb ◦ upstream_response_time_p90:slb ◦ upstream_response_time_p99:slb ◦ upstream_response_time_p9999:slb ◦ request_time_avg:slb ◦ request_time_p50:slb ◦ request_time_p90:slb ◦ request_time_p99:slb ◦ request_time_p9999:slb

Rule name	Status	Algorithm	Metric
slb-patrol-slbid_host_status	This rule is disabled by default.	Time2Graph	◦ pv:slbid:host:status ◦ body_bytes_sent_avg:slbid:host:status ◦ body_bytes_sent_sum:slbid:host:status ◦ request_length_avg:slbid:host:status ◦ request_length_sum:slbid:host:status ◦ upstream_response_time_avg:slbid:host:status ◦ upstream_response_time_p50:slbid:host:status ◦ upstream_response_time_p90:slbid:host:status ◦ upstream_response_time_p99:slbid:host:status ◦ upstream_response_time_p9999:slbid:host:status ◦ request_time_avg:slbid:host:status ◦ request_time_p50:slbid:host:status ◦ request_time_p90:slbid:host:status ◦ request_time_p99:slbid:host:status ◦ request_time_p9999:slbid:host:status

Rule name	Status	Algorithm	Metric
slb-patrol-slbid_host_status_request_method_upstream_status	This rule is disabled by default.	Time2Graph	◦ pv:slbid:host:status:method:upstream_status ◦ body_bytes_sent_avg:slbid:host:status:method:upstream_status ◦ body_bytes_sent_sum:slbid:host:status:method:upstream_status ◦ request_length_avg:slbid:host:status:method:upstream_status ◦ request_length_sum:slbid:host:status:method:upstream_status ◦ upstream_response_time_avg:slbid:host:status:method:upstream_status ◦ upstream_response_time_p50:slbid:host:status:method:upstream_status ◦ upstream_response_time_p90:slbid:host:status:method:upstream_status ◦ upstream_response_time_p99:slbid:host:status:method:upstream_status ◦ upstream_response_time_p9999:slbid:host:status:method:upstream_status ◦ request_time_avg:slbid:host:status:method:upstream_status ◦ request_time_p50:slbid:host:status:method:upstream_status ◦ request_time_p90:slbid:host:status:method:upstream_status ◦ request_time_p99:slbid:host:status:method:upstream_status ◦ request_time_p9999:slbid:host:status:method:upstream_status

- Dedicated dashboards

Dashboard	Associated Logstore or Metricstore	Description
Monitoring Overview	*Logstore for access logs*-metrics	Displays the overall information about the metrics of CLB instances. The metrics include Core Indicators, Error Code, Traffic, Exception Event, PVs, Access Success Rate, and Avg Latency.

Dashboard	Associated Logstore or Metricstore	Description
Monitoring Center	◦ *Logstore for access logs*-metrics ◦ *Logstore for access logs*	Displays the real-time monitoring data of CLB instances. The data includes PVs, Request Success Rate, Average Latency, Requests with Status Code 4xx, Status Distribution, Traffic, P50 Latency, P90 Latency, P99 Latency, P9999 Latency, Hosts with Most Requests, Hosts with Highest Latency, Hosts with Highest Failure Rate, URLs with Most Requests, URLs with Highest Latency, URLs with Highest Failure Rate, Backends with Most Requests, Backends with Highest Latency, and Backends with Highest Failure Rate.
Real-time Monitoring	*Logstore for access logs*	Displays the metrics of CLB instances that are accurate to seconds. The metrics include QPS, Access Latency, Upstream Latency, Success Rate, Request Traffic, Response Body Traffic, Status Code 2xx, Status Code 3xx, Error Codes, Upstream Status Code 2xx, Upstream Status Code 3xx, and Upstream Error Codes.
Instance Inspection	◦ *Logstore for access logs*-metrics ◦ *Logstore for access logs*-metrics-result	Displays information about anomalies that are detected by Log Service. The information includes Exceptions, High Exceptions, Exception Distributions, Middle Exceptions, Low Exceptions, Exception List, and Exception Events. Log Service automatically detects anomalies in CLB instances based on the machine learning algorithms that are provided by Log Service.
Access Overview	*Logstore for access logs*	Displays the overall status of CLB instances. The status information includes PVs (Day-on-day), PVs (Week-on-week), UVs (Day-on-day), UVs (Week-on-week), PV Distribution, UV Distribution, PVs Today, PVs of 7 Days, Top 10 States with Most Requests, Percentage of Mobile Users, TOP 10 Hosts with Most Requests, TOP 10 User Agents with Most Requests, and IP Addresses with Most Requests.

Billing

- You are not charged for the access log management feature of SLB.
- After CLB access logs are collected to Log Service, you are charged based on the storage space, read traffic, number of requests, data transformation, and data shipping. The fees are included in your Log Service bills. For more information, see Billable items.

Limits

- You can enable the collection of access logs only on the CLB instances on which Layer 7 listeners are configured.
- The CLB instance that you use must reside in the same region as the project that you specify in Log Service.

- CloudLens for CLB is supported in the following regions.

Cloud type	Region
Alibaba Cloud public cloud	China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Ulanqab), China (Hangzhou), China (Shanghai), China (Shenzhen), China (Heyuan), China (Guangzhou), China (Chengdu), China (Hong Kong), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), Indonesia (Jakarta), Philippines (Manila), Japan (Tokyo), India (Mumbai), US (Silicon Valley), US (Virginia), Germany (Frankfurt), UK (London), UAE (Dubai), and Russia (Moscow)

Precautions

If you enable a CloudLens application, Log Service automatically checks whether a project whose name is in the aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan format exists within your Alibaba Cloud account. If the project does not exist, Log Service automatically creates the project.

If you want to delete the project, open the Cloud Shell and run the `aliyunlog log delete_project --project_name=aliyun-product-data-<Alibaba Cloud account ID>-cn-heyuan --region-endpoint=cn-heyuan.log.aliyuncs.com` command. Replace Alibaba Cloud account ID based on your business scenario.

> **Notice** If you delete the project, all CloudLens applications become unavailable. Proceed with caution.

12.2. Grant operation permissions on CloudLens for CLB to a RAM user

This topic describes how to grant operation permissions on CloudLens for CLB to a RAM user.

Prerequisites

A RAM user is created. For more information, see Step 1: Create a RAM user.

Context

You can grant operation permissions on CloudLens for CLB to a RAM user in one of the following modes:

- Simple mode: You can grant all permissions on Log Service to the RAM user. You do not need to configure parameters.
- Custom mode: You can create custom policies and attach the policies to the RAM user. This mode allows you to perform fine-grained access control. However, this mode requires complex configurations.

Simple mode

Log on to the RAM console by using your Alibaba Cloud account. Then, attach the AliyunLogFullAccess and AliyunRAMFullAccess policies to the RAM user. This way, the RAM user has all permissions on Log Service. For more information, see Grant permissions to a RAM user.

Custom mode

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Create a policy.
   i. In the left-side navigation pane, choose **Permissions > Policies**.
   ii. On the Policies page, click **Create Policy**.
   iii. On the **Create Policy** page, click the **JSON** tab, replace the existing script in the code editor with one of the following scripts, and then click **Next: Edit Basic Information**.

   You can grant the read-only permissions or read and write permissions on CloudLens for CLB to the RAM user.

   - Read-only permissions: Use the following script to authorize the RAM user only to view each page of CloudLens for CLB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "log:GetProductDataCollection",
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:slb:*:*:loadbalancer/*"
            ],
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

   - Read and write permissions: Use the following script to authorize the RAM user to perform all operations that are supported by CloudLens for CLB.

```
{
    "Statement": [
        {
            "Action": [
                "log:GetLogStore",
```

```
                "log:ListLogStores",
                "log:GetIndex",
                "log:GetLogStoreHistogram",
                "log:GetLogStoreLogs",
                "log:GetDashboard",
                "log:ListDashboard",
                "log:ListSavedSearch",
                "log:CreateLogStore",
                "log:CreateIndex",
                "log:UpdateIndex",
                "log:ListLogStores",
                "log:GetLogStore",
                "log:GetLogStoreLogs",
                "log:CreateDashboard",
                "log:CreateChart",
                "log:UpdateDashboard",
                "log:UpdateLogStore",
                "log:GetProjectLogs"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:log:*:*:project/*/dashboard/*",
                "acs:log:*:*:project/*/savedsearch/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:GetProductDataCollection",
                "log:OpenProductDataCollection",
                "log:CloseProductDataCollection"
            ],
            "Resource": [
                "acs:log:*:*:project/*/logstore/*",
                "acs:slb:*:*:loadbalancer/*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "log:SetGeneralDataAccessConfig"
            ],
            "Resource": [
                "acs:log:*:*:resource/sls.general_data_access.slb.global_conf.sta
ndard_channel/record"
            ],
            "Effect": "Allow"
        },
        {
            "Action": "ram:CreateServiceLinkedRole",
            "Resource": "*",
            "Effect": "Allow",
            "Condition": {
                "StringEquals": {
```

```
                    "ram:ServiceName": "audit.log.aliyuncs.com"
                }
            }
        ],
        "Version": "1"
}
```

iv. Configure the **Name** parameter and click **OK**.

In this example, set the policy name to log-clb-policy.

3. Attach the policy to the RAM user.
   i. In the left-side navigation pane, choose **Identities > Users**.
   ii. On the Users page, find the RAM user to which you want to attach the policy and click **Add Permissions** in the Actions column.
   iii. In the **Select Policy** section of the **Add Permissions** panel, click **Custom Policy**. Then, click the policy that you create in Step . In this example, click log-clb-policy.
   iv. Click **OK**.

12.3. Enable the data collection feature

CloudLens for CLB allows you to enable the data collection feature with a few clicks to collect Classic Load Balancer (CLB) access logs. This topic describes how to enable the data collection feature for CLB instances. This topic also describes the operations that you can perform after you enable the feature.

Prerequisites

- A CLB instance is created. For more information, see Create a CLB instance.
- A Layer 7 listener such as an HTTP or HTTPS listener is configured for the CLB instance. For more information, see Add an HTTP listener or Add an HTTPS listener.
- A project and a Logstore are created in Log Service. For more information, see Create a project and Create a Logstore.

Authorization

Notice You need to perform this operation only once.

1.
2. On the **Cloud Service Lens** tab in the **Log Application** section, click **CloudLens for CLB**.
3. Follow the on-screen instructions to enable CloudLens for CLB.

   When you enable the application, Log Service automatically authorizes CloudLens for CLB to assume the AliyunServiceRolePolicyForSLSAudit service-linked role to collect CLB logs. For more information, see Manage the AliyunServiceRoleForSLSAudit service-linked role.

Enable data collection

1.
2. On the **Cloud Service Lens** tab in the **Log Application** section, click **CloudLens for CLB**.

3. On the **CLB Instance Access** tab of the **Access Management** page, find the CLB instance for which you want to enable data collection and click **Enable**.
4. In the **Enable Access Logs Collect** dialog box, select the project and the Logstore. Then, click **Confirm**.

What to do next

After you enable CloudLens for CLB, you can perform the following operations on the **Access Management** page.

Operation	Description
Manage CLB instances	After you enable CloudLens for CLB, CloudLens for CLB displays all CLB instances within the current Alibaba Cloud account. Click the CLB instance that you want to manage. Then, you are navigated to the SLB console. You can view the details about the CLB instance and create a listener for the instance. For more information, see Overview.
Disable data collection	Find the CLB instance for which you want to disable data collection and click **Disable** in the Access Logs column.
Query and analyze access logs	Find the CLB instance whose access logs you want to query and analyze and click **Access Logs** in the Actions column. Then, you are navigated to the Logstore in which the access logs are stored. You can view, query, and analyze the access logs. For more information, see Query and analyze logs.
Modify the data retention period	On the **Destination Logstore** tab, find the Logstore whose data retention period you want to modify and click the icon.
Use the updated data collection feature	If log shipping is enabled for your CLB instance in the SLB console and you enable log collection in the Log Service console when you use CloudLens for CLB, access logs are automatically collected. However, no anomaly detection or metric extraction is performed. You can click **Upgrade**. Then, CloudLens for CLB automatically detects anomalies, extracts metrics, and generates assets such as detection results, Metricstores, and inspection jobs.

12.4. View data reports

CloudLens for CLB provides the following dashboards to display the information about a Classic Load Balancer (CLB) instance from different dimensions: Monitoring Overview, Monitoring Center, Real-time Monitoring, Instance Inspection, and Access Overview.

Prerequisites

A CLB instance is created, and log collection is enabled for the instance. For more information, see Enable the data collection feature.

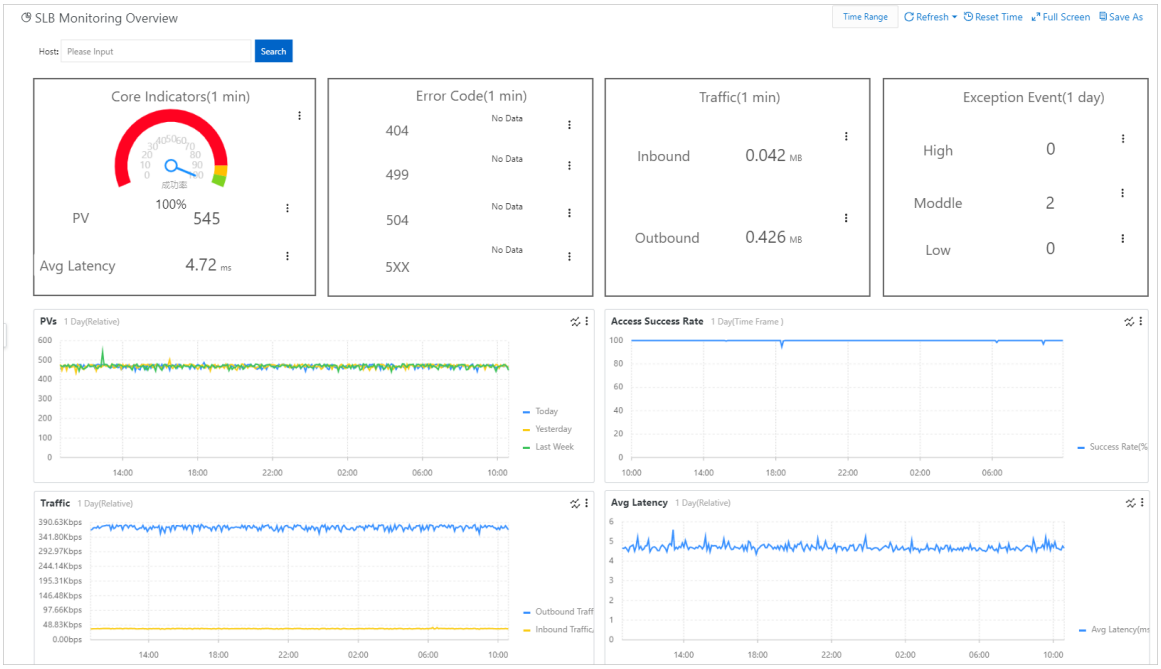
Entry point

1.
2. In the **Log Application** section, click the **Cloud Service Lens** tab. Then, click .

3. In the left-side navigation pane, click **Report Center**.
4. In the upper-left corner of the page, select the CLB instance.

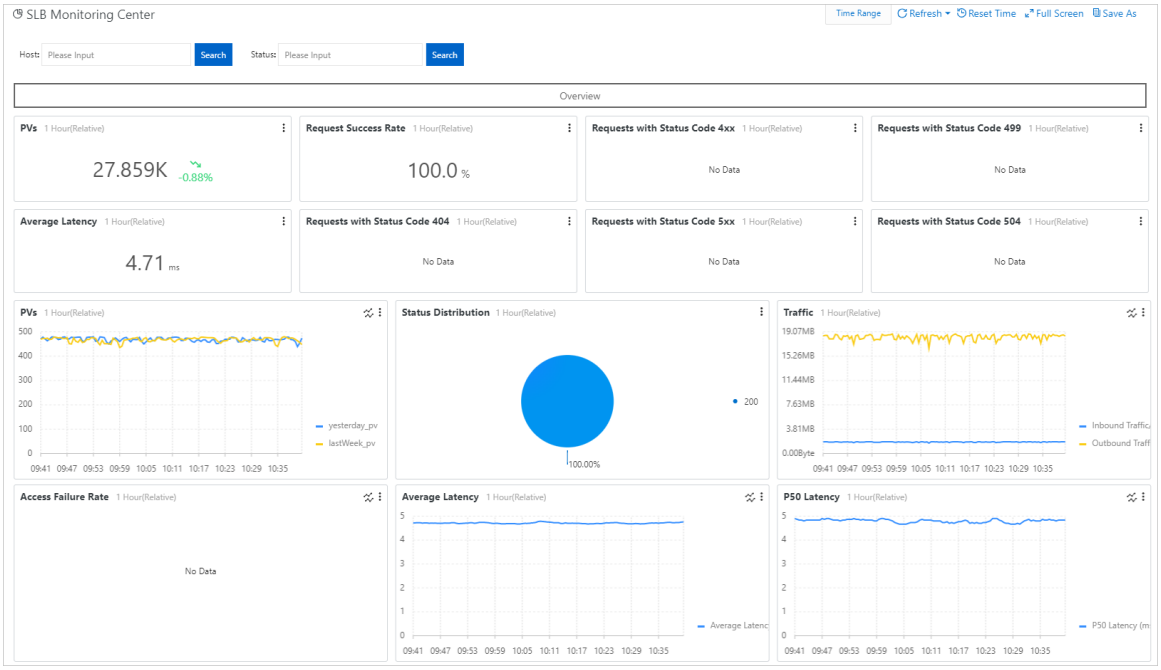
Monitoring Overview

The **Monitoring Overview** dashboard displays the overall information about the metrics of a CLB instance. The metrics include Core Indicators, Error Code, Traffic, Exception Event, PVs, Access Success Rate, and Avg Latency.



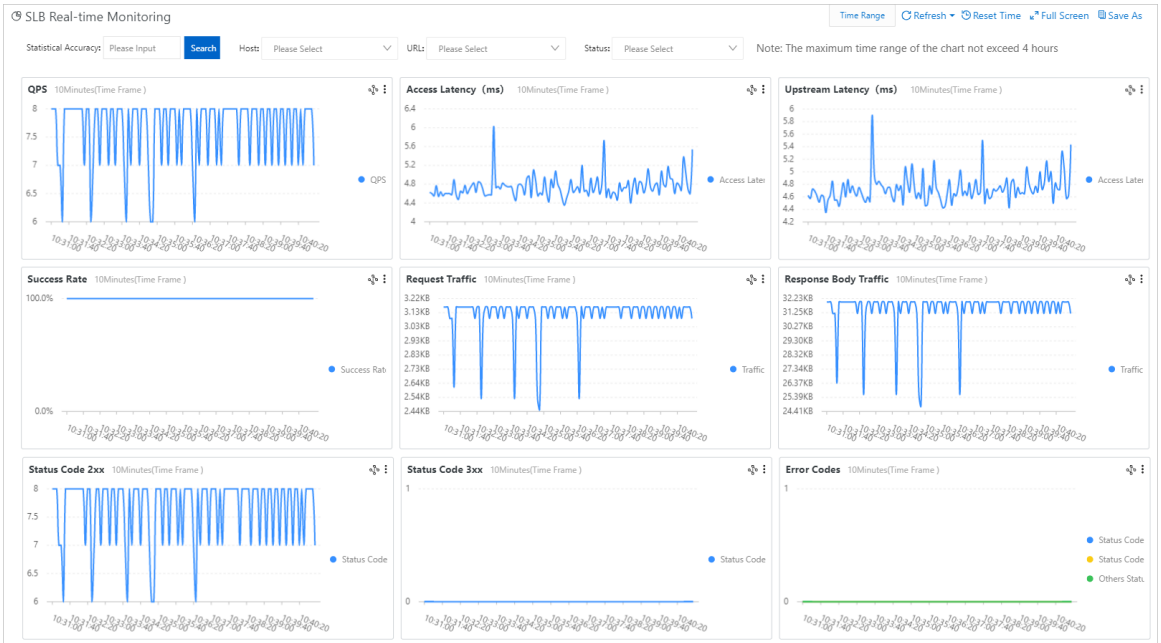
Monitoring Center

The **Monitoring Center** dashboard displays the real-time monitoring data of a CLB instance. The data includes PVs, Request Success Rate, Average Latency, Requests with Status Code 4xx, Status Distribution, Traffic, P50 Latency, P90 Latency, P99 Latency, P9999 Latency, Hosts with Most Requests, Hosts with Highest Latency, Hosts with Highest Failure Rate, URLs with Most Requests, URLs with Highest Latency, URLs with Highest Failure Rate, Backends with Most Requests, Backends with Highest Latency, and Backends with Highest Failure Rate.



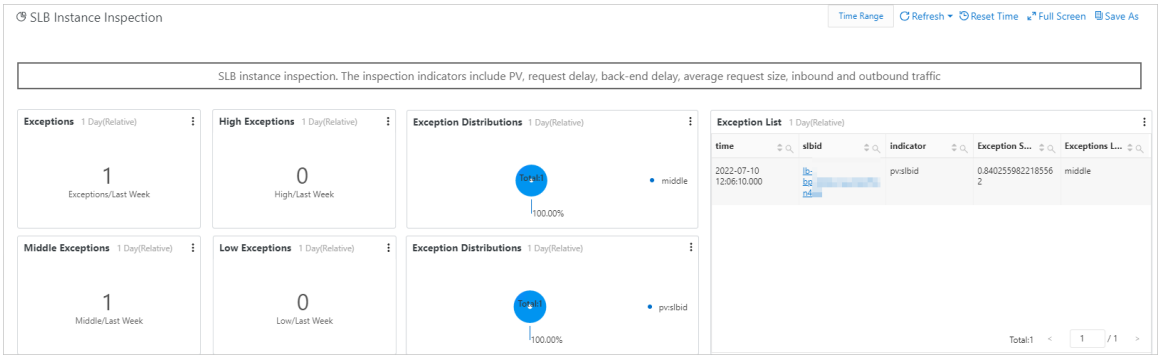
Real-time Monitoring

The **Real-time Monitoring** dashboard displays the metrics of a CLB instance. The metrics are aggregated at one-second intervals and include QPS, Access Latency, Upstream Latency, Success Rate, Request Traffic, Response Body Traffic, Status Code 2xx, Status Code 3xx, Error Codes, Upstream Status Code 2xx, Upstream Status Code 3xx, and Upstream Error Codes.



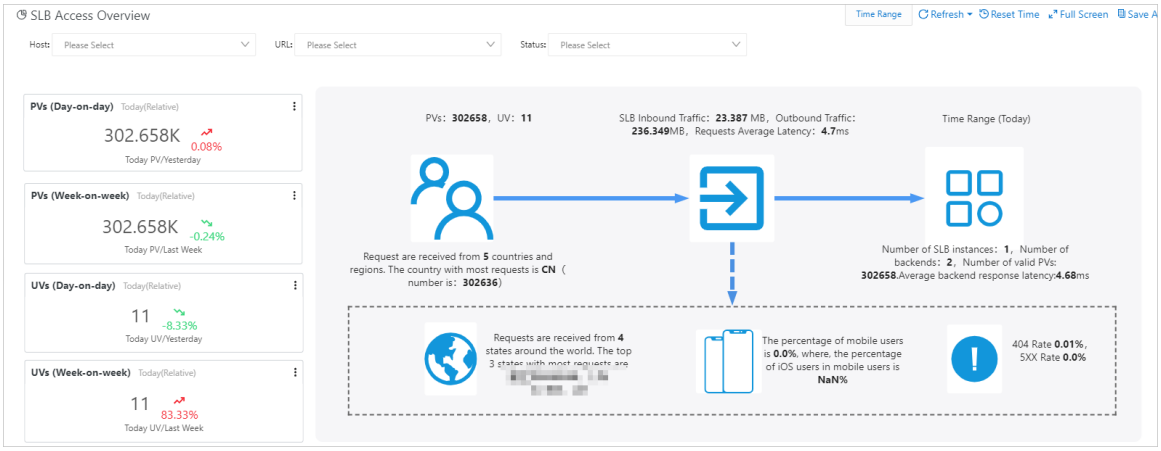
Instance Inspection

The **Instance Inspection** dashboard displays the information about anomalies that are detected by Log Service in a CLB instance. The detection is based on the machine learning algorithms that are provided by Log Service. The information includes Exceptions, High Exceptions, Exception Distributions, Middle Exceptions, Low Exceptions, Exception List, and Exception Events.



Access Overview

The **Access Overview** dashboard displays the status of a CLB instance. The status information includes PVs (Day-on-day), PVs (Week-on-week), UVs (Day-on-day), UVs (Week-on-week), PV Distribution, UV Distribution, PVs Today, PVs of 7 Days, Top 10 States with Most Requests, Percentage of Mobile Users, TOP 10 Hosts with Most Requests, TOP 10 User Agents with Most Requests, and IP Addresses with Most Requests.



12.5. Metrics

This topic describes the metrics that are obtained from the Layer 7 access logs of Classic Load Balancer (CLB). The metrics include global metrics, slbid metrics, status metrics, and upstream_status metrics.

In this topic, the metrics follow the format that is described in Metric. You can query and analyze the metrics by using PromQL or SQL syntax. For more information, see Overview of query and analysis on metrics.

Global metrics

The following table lists global metrics.

Metric	Description
pv	The number of page views (PVs).
body_bytes_sent_avg	The average number of bytes in the bodies of the HTTP responses that are sent to the client.

Metric	Description
body_bytes_sent_sum	The total number of bytes in the bodies of the HTTP responses that are sent to the client.
request_length_avg	The average length of the requests.
request_length_sum	The total length of the requests.
request_time_avg	The average duration of the requests.
request_time_p50	The 50th percentile value for the request durations.
request_time_p90	The 90th percentile value for the request durations.
request_time_p99	The 99th percentile value for the request durations.
request_time_p9999	The 99.99th percentile value for the request durations.
upstream_response_time_avg	The average duration of the request connections. **Note** The upstream_response_time field specifies the interval between the time when the CLB instance connects to the backend server and the time when the CLB instance disconnects from the backend server after the required data is received.
upstream_response_time_p50	The 50th percentile value for the request connection durations.
upstream_response_time_p90	The 90th percentile value for the request connection durations.
upstream_response_time_p99	The 99th percentile value for the request connection durations.
write_response_time_avg	The average time that is taken by the CLB proxy to send the requests to the client after the CLB proxy receives the requests from the backend server. This time is considered as the response time.
write_response_time_p50	The 50th percentile value for the response time.
write_response_time_p90	The 90th percentile value for the response time.
write_response_time_p99	The 99th percentile value for the response time.

slbid metrics

The tag of the slbid metrics is slbid. The following table lists slbid metrics.

Metric	Description
pv:slbid	The number of PVs for an CLB instance.
body_bytes_sent_avg:slbid	The average number of bytes in the bodies of the HTTP responses that are sent to the client.

Metric	Description
body_bytes_sent_sum:slbid	The total number of bytes in the bodies of the HTTP responses that are sent to the client.
request_length_avg:slbid	The average length of the requests.
request_length_sum:slbid	The total length of the requests.
request_time_avg:slbid	The average duration of the requests.
request_time_p50:slbid	The 50th percentile value for the request durations.
request_time_p90:slbid	The 90th percentile value for the request durations.
request_time_p99:slbid	The 99th percentile value for the request durations.
request_time_p9999:slbid	The 99.99th percentile value for the request durations.
upstream_response_time_avg:slbid	The average duration of the request connections. **Note** The upstream_response_time field specifies the interval between the time when the CLB instance connects to the backend server and the time when the CLB instance disconnects from the backend server after the required data is received.
upstream_response_time_p50:slbid	The 50th percentile value for the request connection durations.
upstream_response_time_p90:slbid	The 90th percentile value for the request connection durations.
upstream_response_time_p99:slbid	The 99th percentile value for the request connection durations.
write_response_time_avg:slbid	The average response time.
write_response_time_p50:slbid	The 50th percentile value for the response time.
write_response_time_p90:slbid	The 90th percentile value for the response time.
write_response_time_p99:slbid	The 99th percentile value for the response time.

status metrics

The tag of the status metrics is slbid+host+status. The following table lists status metrics.

Metric	Description
pv:slbid:host:status	The number of PVs for each slbid, host, and status.

Metric	Description
body_bytes_sent_avg:slbid:host:status	The average number of bytes in the bodies of the HTTP responses that are sent to the client.
body_bytes_sent_sum:slbid:host:status	The total number of bytes in the bodies of the HTTP responses that are sent to the client.
request_length_avg:slbid:host:status	The average length of the requests.
request_length_sum:slbid:host:status	The total length of the requests.
request_time_avg:slbid:host:status	The average duration of the requests.
request_time_p50:slbid:host:status	The 50th percentile value for the request durations.
request_time_p90:slbid:host:status	The 90th percentile value for the request durations.
request_time_p99:slbid:host:status	The 99th percentile value for the request durations.
request_time_p9999:slbid:host:status	The 99.99th percentile value for the request durations.
upstream_response_time_avg:slbid:host:status	The average duration of the request connections. **Note** The upstream_response_time field specifies the interval between the time when the CLB instance connects to the backend server and the time when the CLB instance disconnects from the backend server after the required data is received.
upstream_response_time_p50:slbid:host:status	The 50th percentile value for the request connection durations.
upstream_response_time_p90:slbid:host:status	The 90th percentile value for the request connection durations.
upstream_response_time_p99:slbid:host:status	The 99th percentile value for the request connection durations.
write_response_time_avg:slbid:host:status	The average response time.
write_response_time_p50:slbid:host:status	The 50th percentile value for the response time.
write_response_time_p90:slbid:host:status	The 90th percentile value for the response time.

Metric	Description
write_response_time_p99:slbid:host:status	The 99th percentile value for the response time.

upstream_status metrics

The tag of the upstream_status metrics is slbid+host+status+request_method+upstream_status+url. The following table lists upstream_status metrics.

Metric	Description
pv:slbid:host:status:method:upstream_status	The number of PVs for each slbid, host, status, method, url, and upstream_status.
body_bytes_sent_avg:slbid:host:status:method:upstream_status	The average number of bytes in the bodies of the HTTP responses that are sent to the client.
body_bytes_sent_sum:slbid:host:status:method:upstream_status	The total number of bytes in the bodies of the HTTP responses that are sent to the client.
request_length_avg:slbid:host:status:method:upstream_status	The average length of the requests.
request_length_sum:slbid:host:status:method:upstream_status	The total length of the requests.
request_time_avg:slbid:host:status:method:upstream_status	The average duration of the requests.
request_time_p50:slbid:host:status:method:upstream_status	The 50th percentile value for the request durations.
request_time_p90:slbid:host:status:method:upstream_status	The 90th percentile value for the request durations.
request_time_p99:slbid:host:status:method:upstream_status	The 99th percentile value for the request durations.
request_time_p9999:slbid:host:status:method:upstream_status	The 99.99th percentile value for the request durations.
upstream_response_time_avg:slbid:host:status:method:upstream_status	The average duration of the request connections. **Note** The upstream_response_time field specifies the interval between the time when the CLB instance connects to the backend server and the time when the CLB instance disconnects from the backend server after the required data is received.
upstream_response_time_p50:slbid:host:status:method:upstream_status	The 50th percentile value for the request connection durations.

Metric	Description
upstream_response_time_p90:slbid:host:status:method:upstream_status	The 90th percentile value for the request connection durations.
upstream_response_time_p99:slbid:host:status:method:upstream_status	The 99th percentile value for the request connection durations.
write_response_time_avg:slbid:host:status:method:upstream_status	The average response time.
write_response_time_p50:slbid:host:status:method:upstream_status	The 50th percentile value for the response time.
write_response_time_p90:slbid:host:status:method:upstream_status	The 90th percentile value for the response time.
write_response_time_p99:slbid:host:status:method:upstream_status	The 99th percentile value for the response time.

12.6. Log fields

This topic describes the fields in the Layer 7 access logs of Classic Load Balancer (CLB).

Field	Description
__topic__	The topic of the log. The value is fixed as slb_layer7_access_log.
body_bytes_sent	The number of bytes in the body of the HTTP response that is sent to the client.
client_ip	The IP address of the client.
host	The host. By default, the value is obtained from request parameters. If no value is obtained from request parameters, the value is obtained from the Host header. If no value is obtained from request parameters or the Host header, the IP address of the backend server that processes the request is used as the value.
http_host	The Host header of the HTTP request.
http_referer	The Referer header of the HTTP request that is received by the proxy.
http_user_agent	The User-Agent header of the HTTP request that is received by the proxy.
http_x_forwarded_for	The X-Forwarded-For (XFF) header of the HTTP request that is received by the proxy.
http_x_real_ip	The originating IP address of the client.

Field	Description
read_request_time	The time that is taken by the proxy to read the request. Unit: milliseconds.
request_length	The length of the request. The request line, request headers, and request body are all counted.
request_method	The request method.
request_time	The interval between the time when the proxy receives the first request and the time when the proxy returns a response. Unit: seconds.
request_uri	The URI of the request that is received by the proxy.
scheme	The schema of the request. Valid values: HTTP and HTTPS.
server_protocol	The HTTP version of the request that is received by the proxy. Examples: HTTP/1.0 and HTTP/1.1.
slb_vport	The listening port of the CLB instance.
slbid	The ID of the CLB instance.
ssl_cipher	The cipher suite that is used to establish an SSL connection. Example: ECDHE-RSA-AES128-GCM-SHA256.
ssl_protocol	The protocol that is used to establish an SSL connection. Example: TLSv1.2.
status	The HTTP status code that is sent by the proxy.
tcpinfo_rtt	The round-trip time (RTT) of the TCP connection that is established by the client. Unit: microseconds.
time	The time when the log is generated.
upstream_addr	The IP address and port number of the backend server.
upstream_response_time	The interval between the time when the CLB instance connects to the backend server and the time when the CLB instance disconnects from the backend server after the required data is received. Unit: seconds.
upstream_status	The HTTP status code that is received by the proxy from the backend server.
vip_addr	The virtual IP address.
write_response_time	The time that is taken by the CLB proxy to send the request to the client after the CLB proxy receives the request from the backend server.

13.Cost Manager

13.1. Cost Manager

Log Service provides the Cost Manager application. After you enable the application, bills are automatically imported to the application. You can view the dashboards of bills in the application. This makes bill analysis more efficient.

Context

Alibaba Cloud provides highly elastic resources and offers a variety of specifications. You can use the resources at any time based on your business requirements. However, you need to take note of the costs for the resources that you use. Alibaba Cloud provides the pay-as-you-go and subscription billing methods. For the bills of pay-as-you-go resources, manual statistical analysis is time-consuming and error-prone. To resolve this issue, Log Service provides the Cost Manager application. You can use this application to collect and sort bills. This makes bill analysis more efficient.

Features

After you enable the application, bills are automatically imported from the billing center to the Logstore that you specify. A bill is a type of time series data. Log Service can collect, store, and analyze time series data in real time. If you use Log Service to analyze bills, you can reduce labor costs for bill analysis by 80%. The Cost Manager application provides the following features:

- Near-real-time bill collection: The application uploads bills to Log Service within one hour after the bills are generated.
- Custom dashboards: The application provides a variety of capabilities to meet the requirements for common analysis scenarios and automatically sends reports.
- Interactive analysis: You can specify query statements to analyze bills. The analysis results are returned in seconds. You can save the query statements into charts on custom dashboards.
- Visualization: The application displays analysis results in charts, which is more intuitive.
- Machine learning algorithms: The application forecasts bill trends and identifies unusual bills.
- Custom alerts: The application supports custom alert settings. This allows you to learn about bill details in real time.
- Free storage and analysis: You are not charged for the storage and analysis of bills.

Import bills

1.
2. In the **Log Application** section, click the **Business Analysis** tab. Then, click **Cost Manager**.
3. In the left-side navigation pane, click **Settings**.
4. Import your bill and click **Next**.

   In the Import Bills step, perform the following operations:

   - **Import Alibaba Cloud Bills**: If you select this option, bills of all Alibaba Cloud services within the current account are imported to Log Service.
   - **First Import of History Bills**: If you import a billing history for the first time, you can specify a time range for the billing history.
   - Obtain permissions to access bills: If the current account is not granted the permissions to access

bills, you can follow the on-screen instructions to grant the permissions to the account.

5. Configure a subscription and click **Update Resources & Next**.

   In the Subscribe to Reports step, configure the following parameters:

   - **Frequency**: the frequency at which subscribed reports are sent.
   - **Add Watermark**: If you turn on Add Watermark, bills are watermarked to prevent the leakage of sensitive data.
   - **Subscription**: If you turn on **Subscription**, you can subscribe to reports.
   - **Notifications**: You can select **Email** or **WebHook-DingTalk Bot** from the Notifications drop-down list to send subscribed reports. For more information about how to obtain the webhook URL of a DingTalk chatbot, see DingTalk chatbot webhooks.

6. (Optional)Configure alerts and click **Complete**.

   A new version of the alerting feature is available. You can click **Alerts** to configure alerts by using the new alerting feature. For more information about the parameters, see Configure an alert monitoring rule in Log Service.

Feature description

After you import bills, you can choose **Cost Manager > Introduction** in the left-side navigation pane to view an introduction to the Cost Manager application. The introduction includes the description, usage guide, and limits. The descriptions for the fields in bills are also provided.

Custom Analysis

On the Custom Analysis page, you can query and analyze the bills that you import, configure query statements as saved searches, save analysis results as charts on dashboards, and configure alerts.

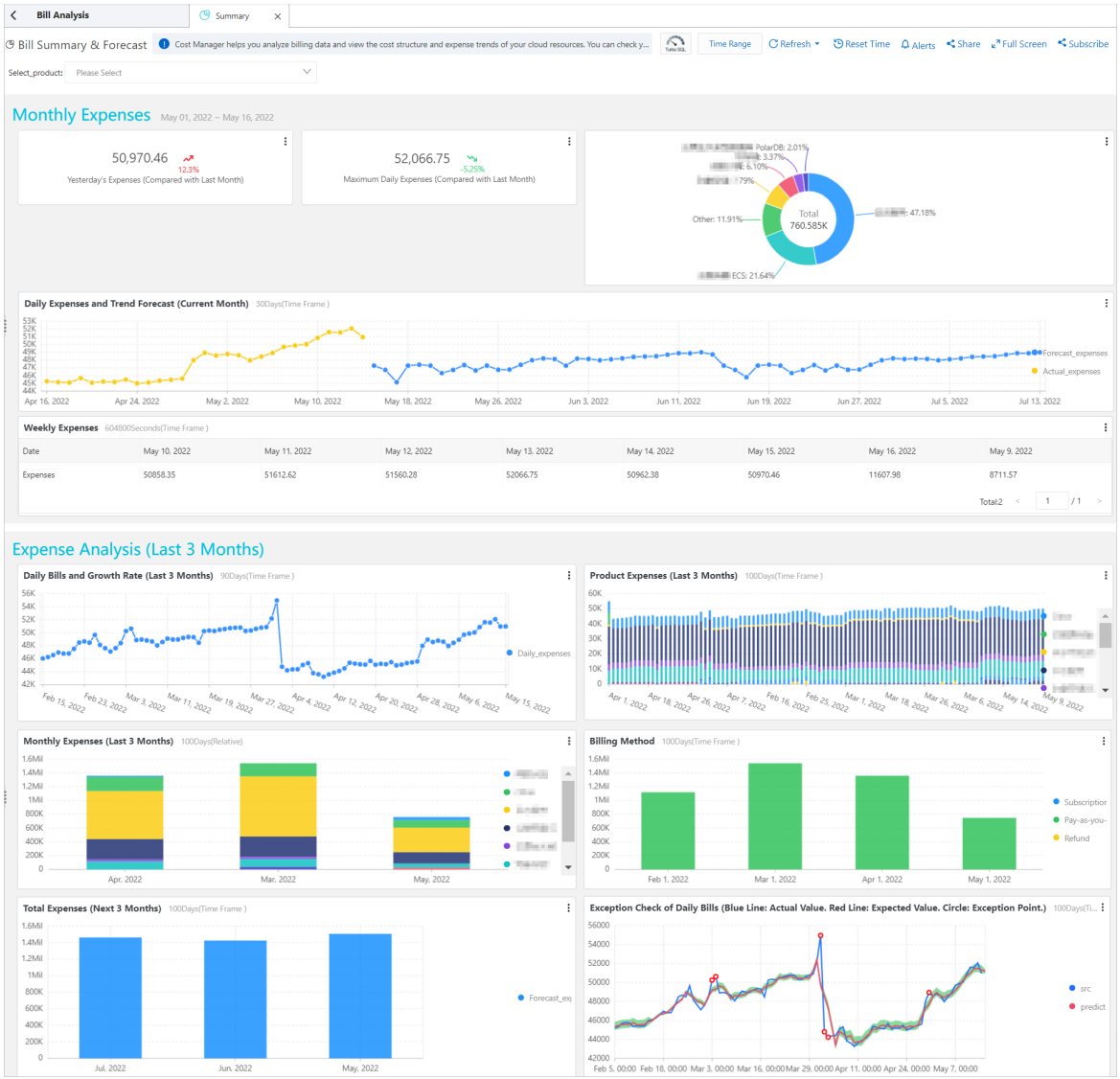
1. In the left-side navigation pane, choose **Cost Manager > Custom Analysis**.
2. On the Custom Analysis page, enter a query statement in the search box to query and analyze the bills that you import.

   You can query and analyze bills in the same manner as you query and analyze logs in a Logstore. For more information, see Query and analyze logs.

Summary

The Cost Manager application provides a built-in Summary dashboard to display overall information about bills. The dashboard displays the fees generated by services within your account in the current month and the fees generated by services within your account over the last three months. The dashboard also displays a forecast trend of fees based on the fees of the current month. You can use the trend chart to plan your budgets. The Summary dashboard functions the same as Log Service dashboards. For more information, see 可视化概述.

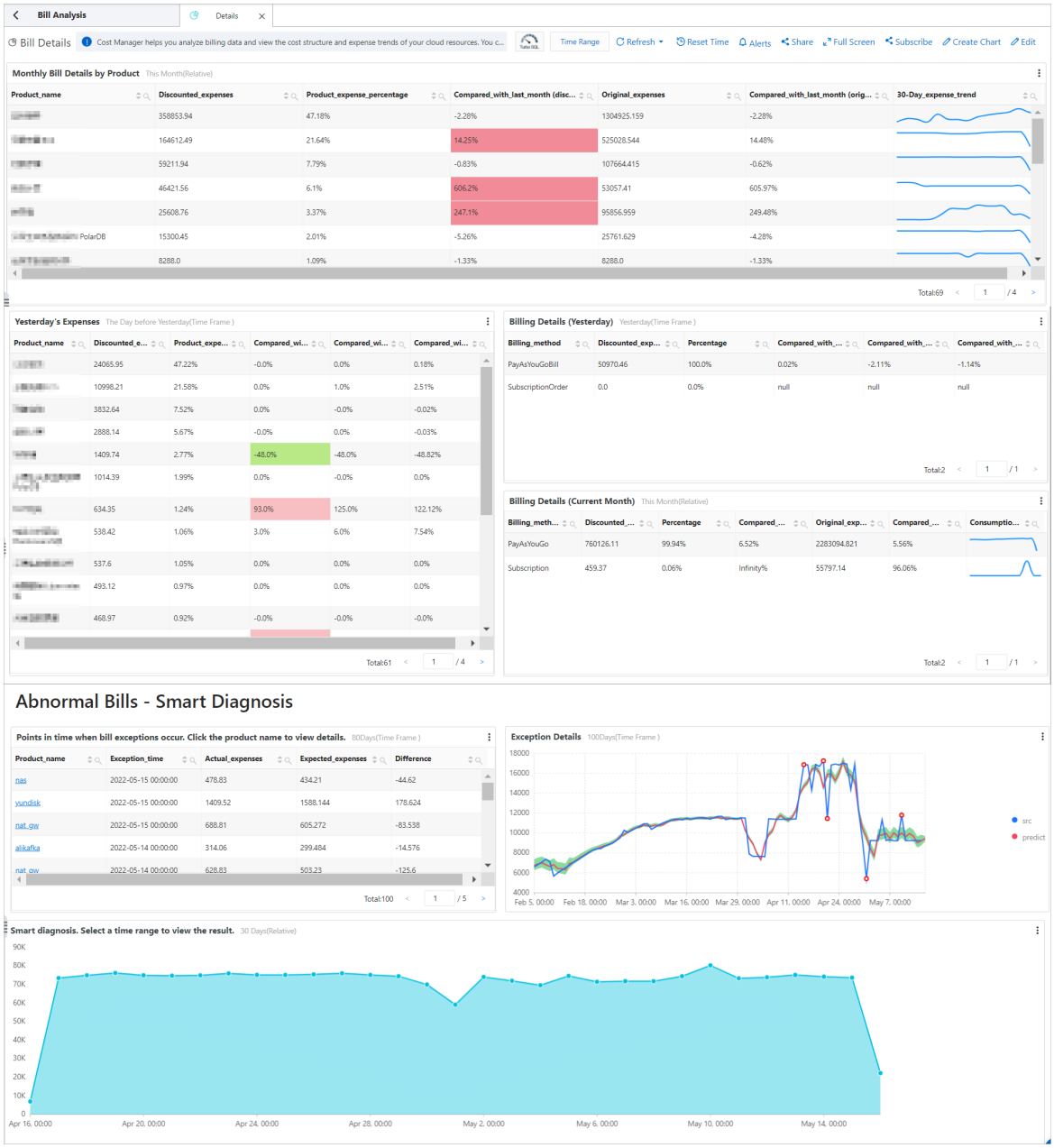
1. In the left-side navigation pane, choose **Cost Manager > Summary**.
2. View the bill overview and forecast information.



Details

The Cost Manager application provides a built-in Details dashboard to display the bill details and trends of each service, and unusual bills. The Details dashboard functions the same as Log Service dashboards. For more information, see 可视化概述.

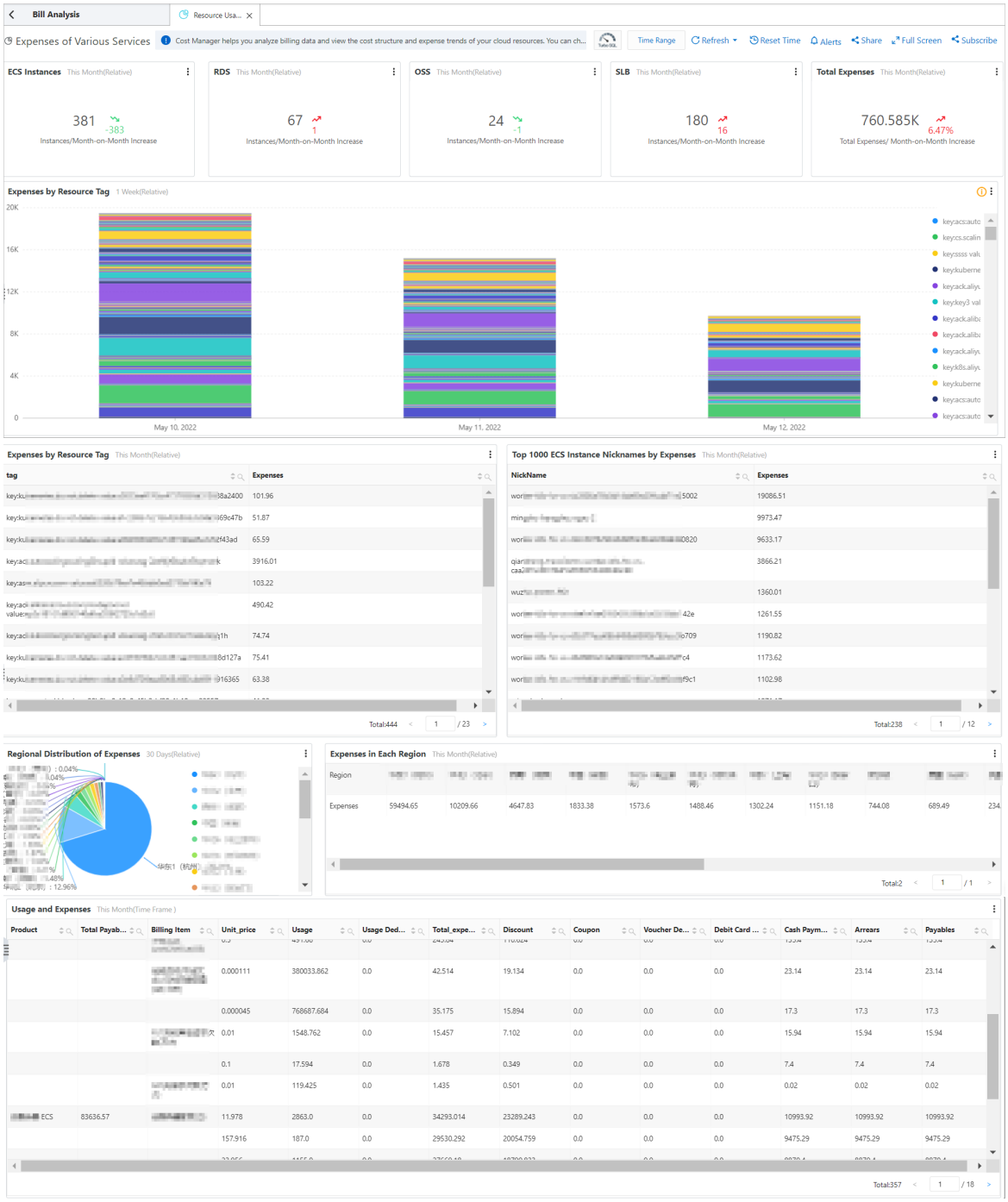
1. In the left-side navigation pane, choose **Cost Manager > Details**.
2. View the consumption details of your services.



Resource Usage Overview

The Resource Usage Overview dashboard displays the number of cloud resources that you use. You can manage the costs of resources by tag or alias.

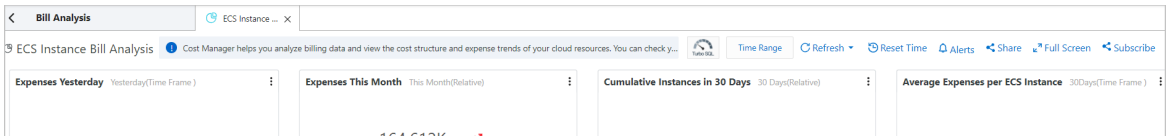
1. In the left-side navigation pane, choose **Cost Manager > Resource Usage Overview**.
2. View charts on the Resource Usage Overview dashboard.

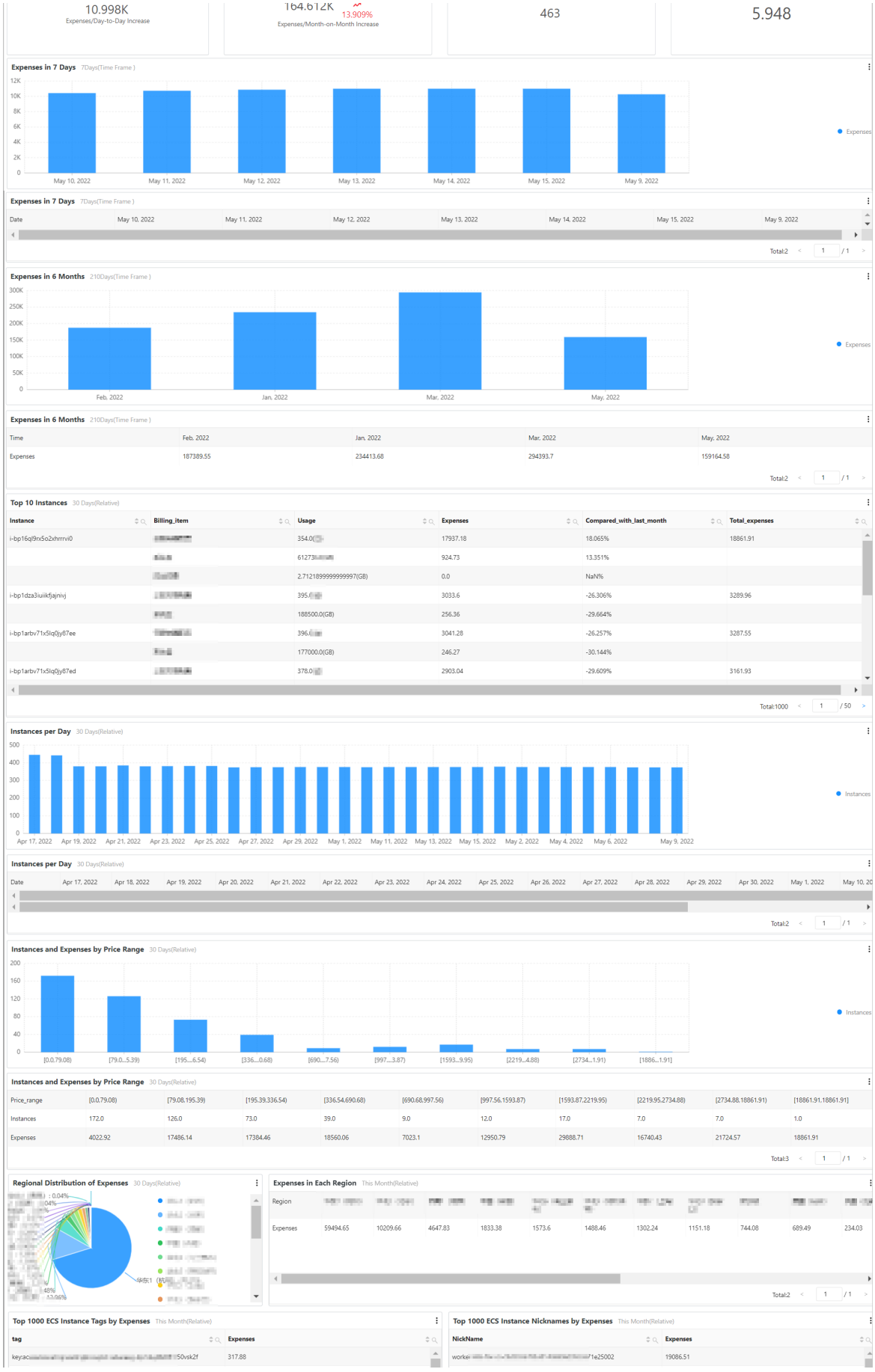


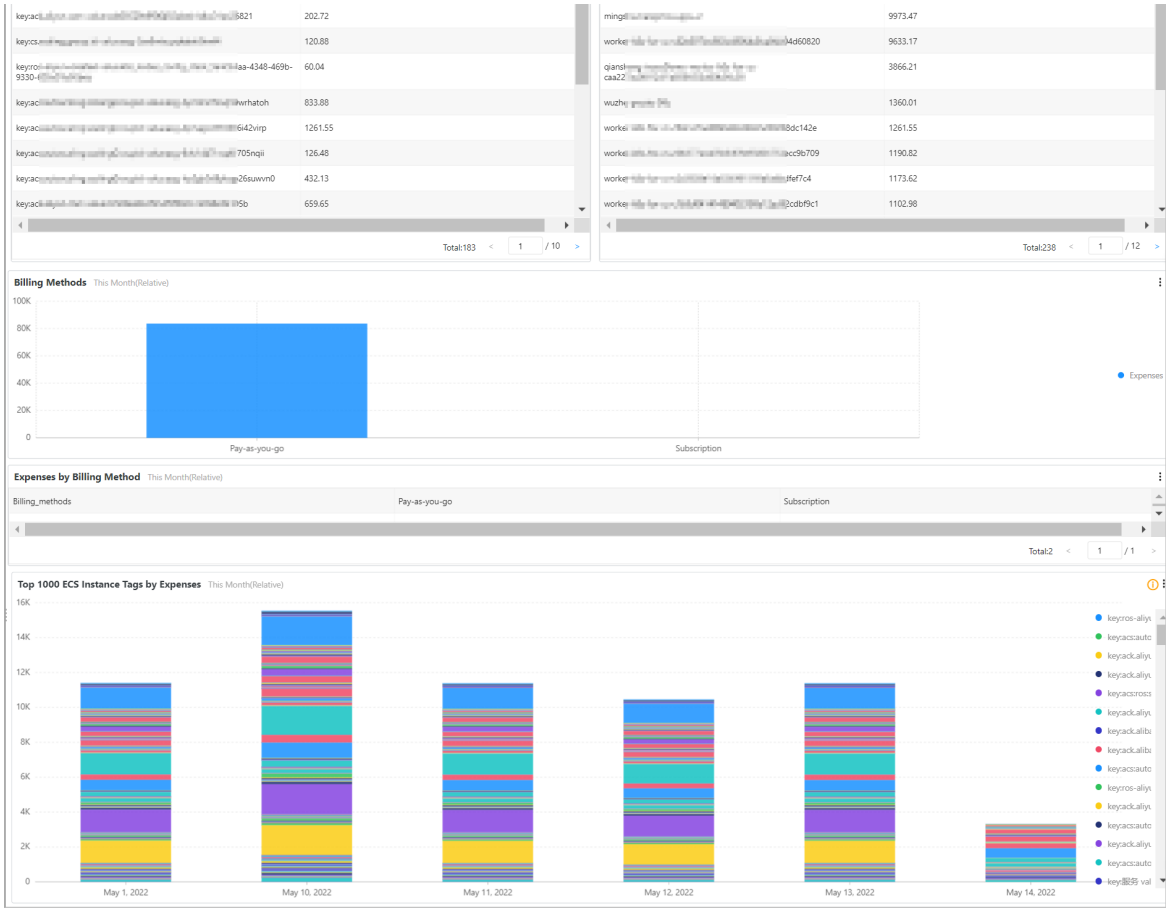
ECS Instance Bill Analysis

The ECS Instance Bill Analysis dashboard provides a usage overview of ECS instances. You can analyze the bills of ECS instances by region, tag, and alias. We recommend that you use the ECS Instance Bill Analysis dashboard to plan budgets and allocate costs.

1. In the left-side navigation pane, choose **Cost Manager > ECS Instance Bill Analysis**.
2. View charts on the ECS Instance Bill Analysis dashboard.



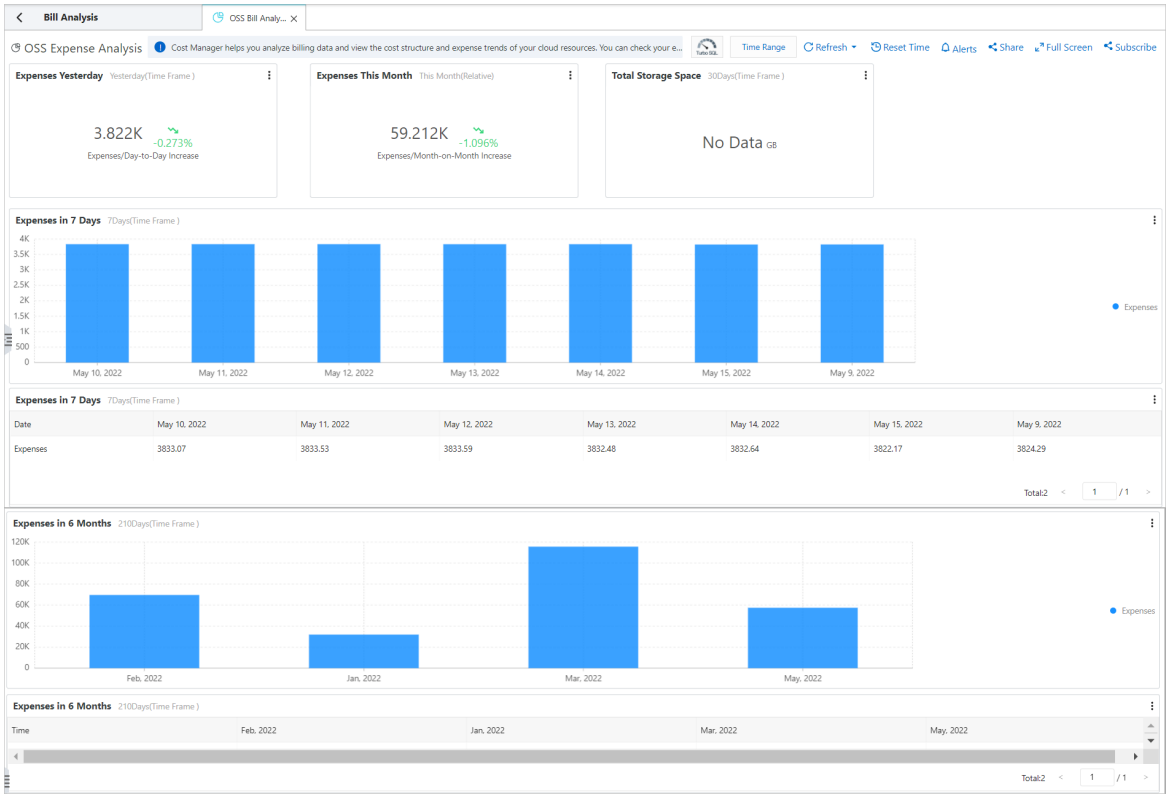

10.998K
Expenses/Day-to-Day Increase
164.612K
13.909%
Expenses/Month-on-Month Increase
463
5.948
Expenses in 7 Days
Expenses in 6 Months
Top 10 Instances
Instances per Day
Instances and Expenses by Price Range
Regional Distribution of Expenses
Expenses in Each Region
Top 1000 ECS Instance Tags by Expenses
Top 1000 ECS Instance Nicknames by Expenses



OSS Bill Analysis

The OSS Bill Analysis dashboard displays the total cost of OSS, cost trends, storage costs of OSS objects for different storage classes, and usage and cost of each billable item. The storage classes include Standard, Infrequent Access, and Archive. You can change the storage class that you use based on your business requirements to reduce costs.

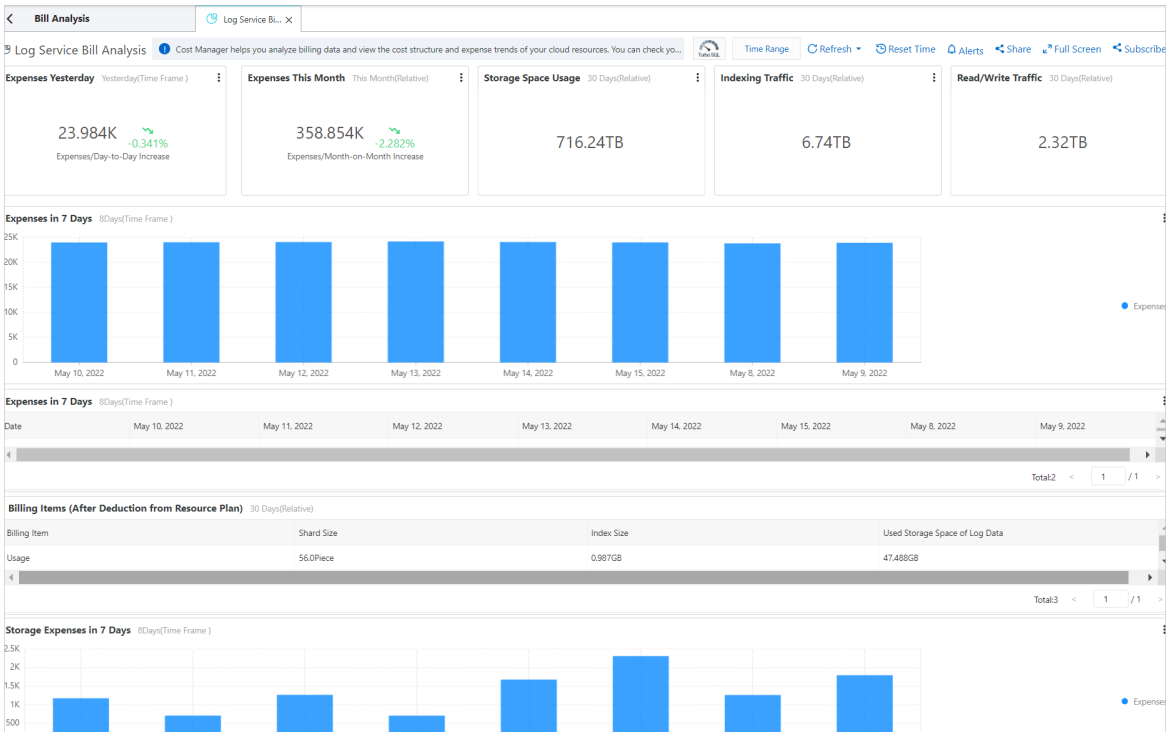
1. In the left-side navigation pane, choose **Cost Manager > OSS Bill Analysis**.
2. View charts on the OSS Bill Analysis dashboard.



Log Service Bill Analysis

The Log Service Bill Analysis dashboard displays the total cost of Log Service, cost trends, usage of each billable item, and projects and Logstores that generate the highest fees for storage and index traffic. You can use the dashboard to reduce the cost of Log Service.

1. In the left-side navigation pane, choose **Cost Manager > Log Service Bill Analysis**.
2. View charts on the Log Service Bill Analysis dashboard.




Storage Usage in 7 Days 8Days(Time Frame)
Indexing Expenses in 7 Days 8Days(Time Frame)
Indexing Traffic in 7 Days 8Days(Time Frame)
Expenses by Read/Write Traffic in 7 Days 8Days(Time Frame)
Read/Write Traffic in 7 Days 8Days(Time Frame)
Expenses by Read/Write Times in 7 Days 8Days(Time Frame)
Read/Write Times in 7 Days 8Days(Time Frame)
Top Project 8Days(Time Frame)
Top Logstore by Expenses 8Days(Time Frame)

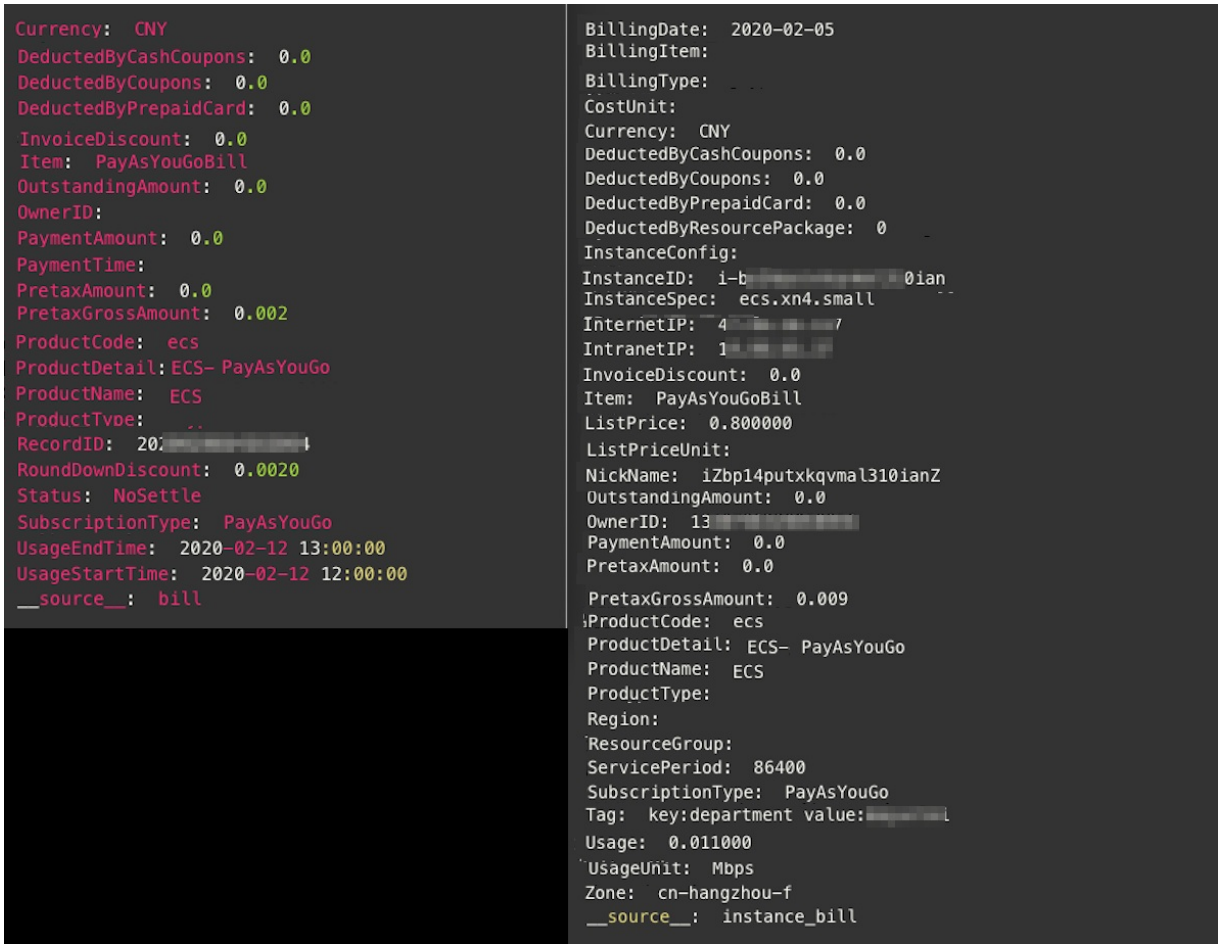
13.2. Use SQL statements to analyze bills

This topic describes how to use SQL statements to analyze bills in the Log Service console.

Billing data details

Billing data includes the following two categories:

- Billing data of a service. This category of billing data is indicated by the field `source:bill`, as shown in the left section of the following figure. A billing record is generated for a service in every billing cycle.
- Billing data of an instance. This category of billing data is indicated by the field `source:instance_bill`, as shown in the right section of the following figure. A billing record is generated for an instance in every billing cycle. The billing data includes the information about the usage, attributes such as tag, alias, and name, and costs of the instance.

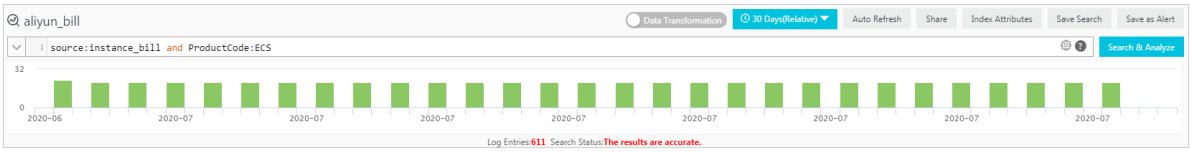


Examples

The built-in reports in the Cost Manager application are templates for data analysis. To satisfy your various requirements, you can customize SQL statements to analyze bills. Bills of ECS instances are used as an example to show how to use SQL statements for bill analysis.

- Query bills that are crucial to your business

To query bills of ECS instances that are crucial to your business in the Logstore named aliyun_bill, enter `source:instance_bill and ProductCode:ECS` in the search box. The following figure shows the query results. For more information about query syntax, see Search syntax.



- Aggregate bills

  Execute the following SQL statement to retrieve the total cost of all ECS instances. On the Graph tab, click **Add to New Dashboard**. A dedicated dashboard is created for the bills.

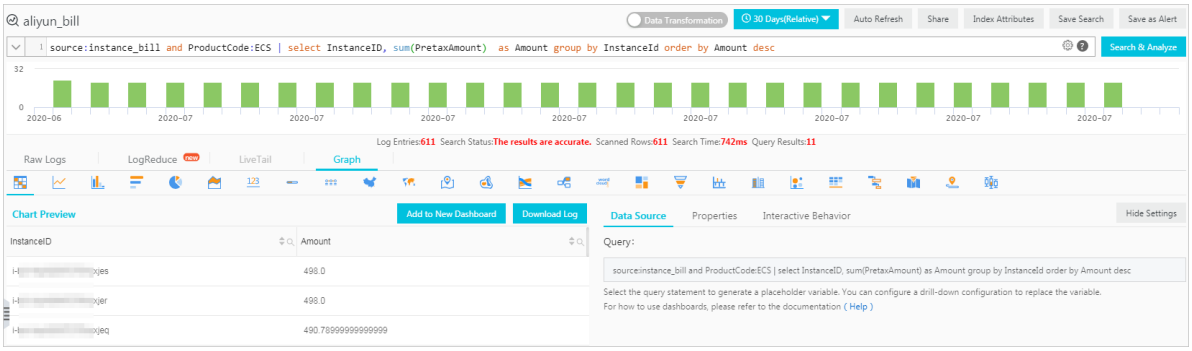
```
source:instance_bill and ProductCode:ECS | select sum(PretaxAmount)
```

- Group bills

  Execute the following SQL statement to retrieve the total cost of each ECS instance:

```
source:instance_bill and ProductCode:ECS | select InstanceID, sum(PretaxAmount)  as Amoun
t group by InstanceId order by Amount desc
```

  The bills of ECS instances are queried based on their instance IDs. To query the bills of ECS instances based on other attributes such as region or alias, replace the attribute in the `group by` clause with the target attribute.



- Perform month-on-month analysis
  - Calculate the costs of ECS instance for this month and the month-on-month growth rate.

```
source:bill | select diff[1] as "costs of this month", diff[2] as "costs of the previou
s month", difference [3]* 100-100 as "month-on-month growth%" from (select compare(amou
nt,604800) as diff from (select sum(PretaxAmount) as amount from log ))
```

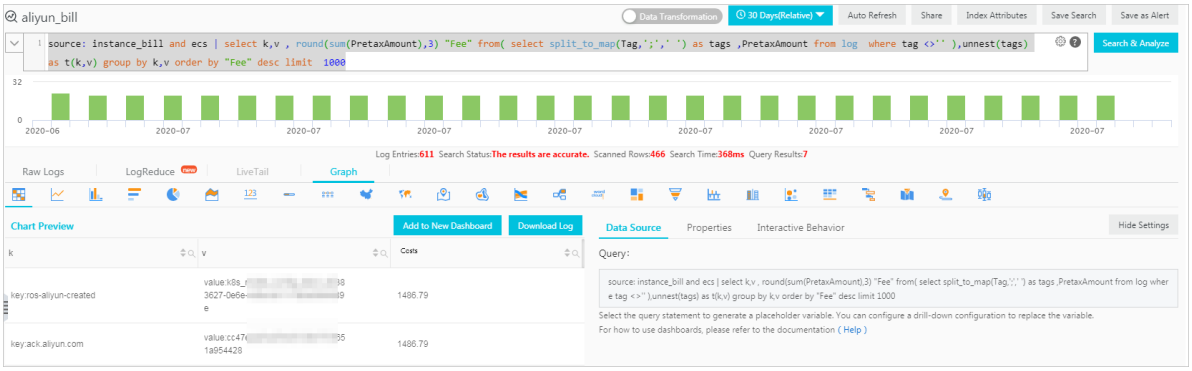
  - Perform month-on-month analysis based on the product code

```
source:bill | select ProductCode, diff[1] as "costs of this month", diff[2] as "costs o
f the previous month", diff[3]* 100-100 as "month-on-month growth%" from (select produc
tcode, compare(amount,604800) as diff from (select ProductCode, sum(PretaxAmount) as am
ount from log group by ProductCode ) group by productcode)
```

- Categorize bills by tag

  You can use tags to categorize bills for multiple services. A tag contains one or more key-value pairs. You can parse a key-value pair of an instance to calculate the costs of the instance.

```
source: instance_bill and ecs | select k,v , round(sum(PretaxAmount),3) "Costs" from( sel
ect split_to_map(Tag,';', '') as tags, pretaxAmount from log where tag <>'' ),unnest(tags
) as t(k,v) group by k,v order by "Costs" desc limit 1000
```



13.3. Authorize a RAM user to use Cost Manager

This topic describes how to authorize a RAM user to use the Cost Manager application.

For more information about how to authorize a RAM user, see Step 2: Grant permissions to the RAM user.

To authorize a RAM user to use the Cost Manager application, use the following policy. For more information about relevant actions, see Action list.

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": "log:CreateLogStore",
      "Resource": "acs:log:*:*:project/bill-analysis-*/logstore/*",
      "Effect": "Allow"
    },
    {
      "Action": "log:CreateIndex",
      "Resource": "acs:log:*:*:project/bill-analysis-*/logstore/aliyun_bill",
      "Effect": "Allow"
    },
    {
      "Action": "log:UpdateIndex",
      "Resource": "acs:log:*:*:project/bill-analysis-*/logstore/aliyun_bill",
      "Effect": "Allow"
    },
    {
      "Action": "log:CreateDashboard",
      "Resource": "acs:log:*:*:project/bill-analysis-*/dashboard/*",
      "Effect": "Allow"
    },
  {
      "Action": "log:UpdateDashboard",
      "Resource": "acs:log:*:*:project/bill-analysis-*/dashboard/*",
      "Effect": "Allow"
    }
```

```
    },
  {
      "Action": "log:CreateSavedSearch",
      "Resource": "acs:log:*:*:project/bill-analysis-*/savedsearch/*",
      "Effect": "Allow"
    },
  {
      "Action": "log:UpdateSavedSearch",
      "Resource": "acs:log:*:*:project/bill-analysis-*/savedsearch/*",
      "Effect": "Allow"
    },
{
      "Action": "log:CreateJob",
      "Resource": "acs:log:*:*:project/bill-analysis-*/job/*",
      "Effect": "Allow"
    },
  {
      "Action": "log:UpdateJob",
      "Resource": "acs:log:*:*:project/bill-analysis-*/job/*",
      "Effect": "Allow"
    },
 {
      "Action": "log:CreateApp",
      "Resource": "acs:log:*:*:app/bill",
      "Effect": "Allow"
    },
{
      "Action": "log:UpdateApp",
      "Resource": "acs:log:*:*:app/bill",
      "Effect": "Allow"
    },
{
      "Action": "log:GetApp",
      "Resource": "acs:log:*:*:app/bill",
      "Effect": "Allow"
    },
{
      "Action": "log:DeleteApp",
      "Resource": "acs:log:*:*:app/bill",
      "Effect": "Allow"
    }
  ]
}
```

14.Analysis of the epidemic situation of new Crown virus

14.1. Overview

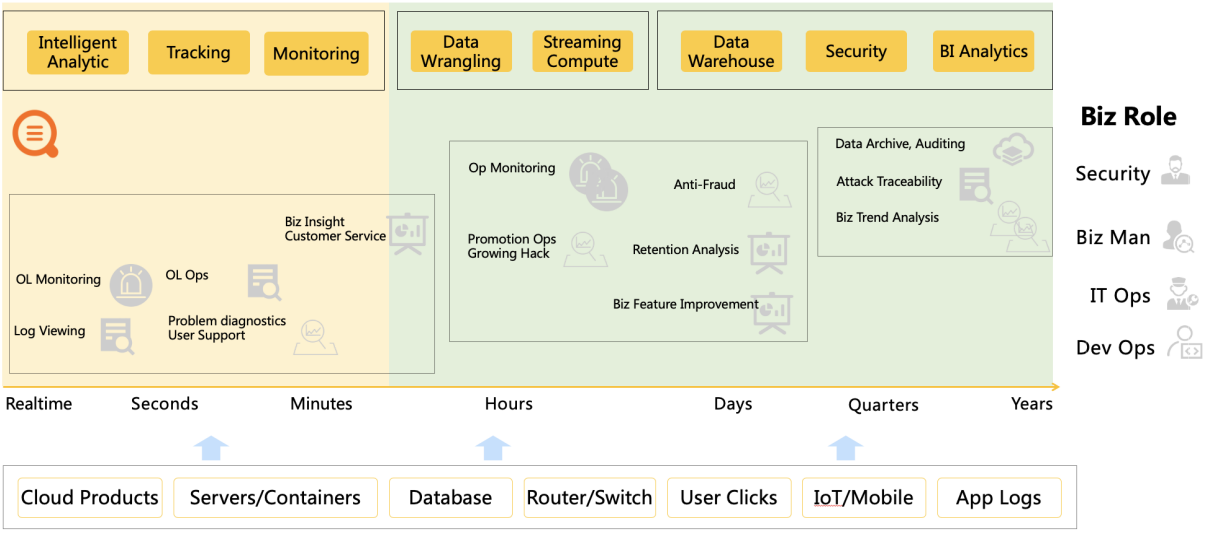
This topic describes the novel coronavirus (COVID-19) pandemic analysis application and its features.

Background information

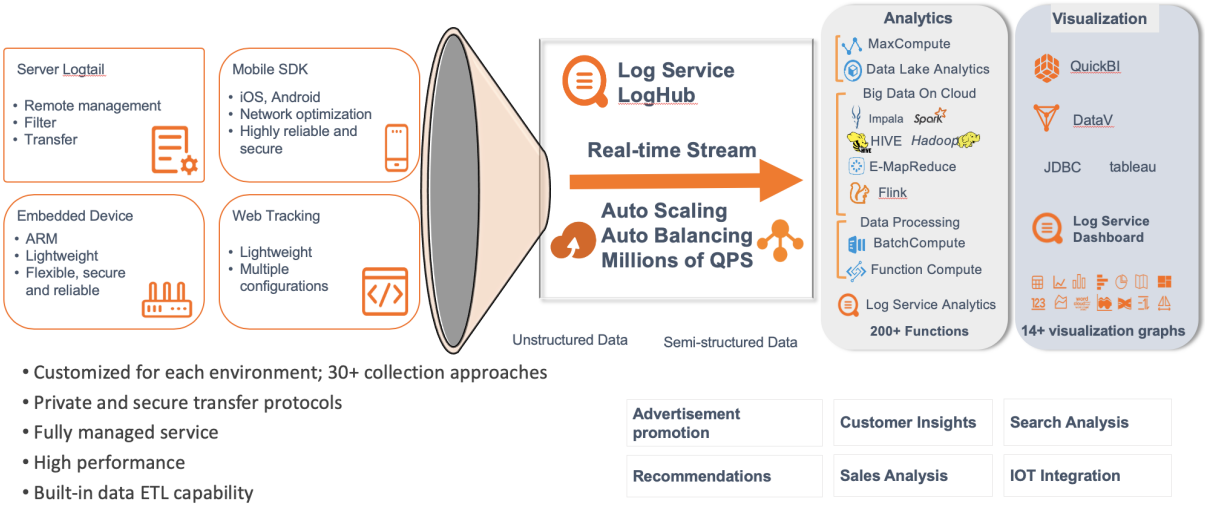
The COVID-19 pandemic analysis application is designed based on the visualized data search and analytics feature of Alibaba Cloud Log Service. You can use the COVID-19 pandemic analysis application to analyze the impact of COVID-19 on various countries, regions, provinces, and states. The COVID-19 pandemic analysis application is available to governments, communities, third-party platforms, and developers. For more information, see Application operation and management.

What is Log Service?

Log Service is a one-stop service to process log data. The service allows you to collect, consume, ship, search, and analyze large amounts of log data without the need for extra code resources. This helps you improve O&M and operational efficiency. Log Service provides features such as real-time data collection, consumption, shipping, search, and analysis. Log Service facilitates real-time monitoring, and is applicable to development, O&M, operations, and security control of data warehouses and other systems.



As the log analysis mid-end, Log Service supports one-stop data collection, processing, search, analysis, AI-based computing, and data visualization. Log Service can also be integrated with other services.



Features

- Synchronizes data about the COVID-19 pandemic in countries, regions, provinces, and states across the globe on a regular basis, and visualizes the analysis results.

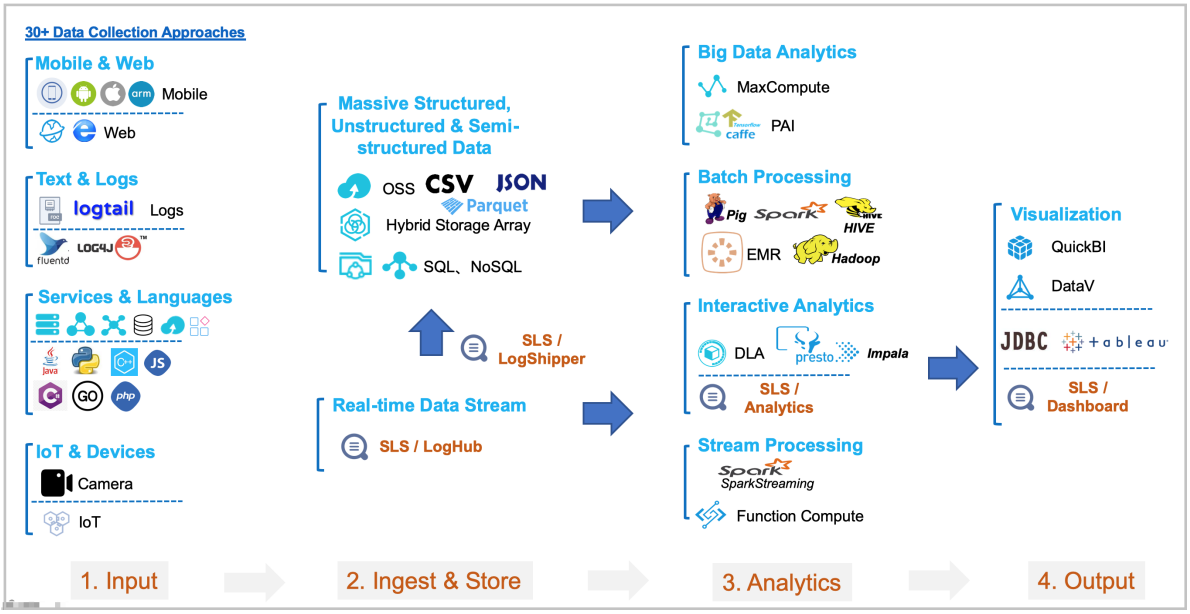
  > **Note** Log Service allows you to collect data from various data sources, as shown in the preceding figure. The COVID-19 pandemic data in Log Service comes from the data repository of the 2019 Novel Coronavirus Visual Dashboard operated by the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE).

- Provides multiple built-in dashboards and allows you to customize dashboards.

  Log Service provides built-in dashboards that display the COVID-19 pandemic data of countries, regions, provinces, and states across the globe. By using Log Service, you can search and analyze the pandemic data, use charts to visualize query results, and organize the charts in dashboards. You can also configure drill-down or drill-up events and alerts.

- Supports connection to various data sources

  Log Service can be integrated with other Alibaba Cloud services, third-party services, open-source platforms, and other systems. For example, Log Service can collect data from DataV, Blink, OSS, Realtime Compute, Grafana, and SOC. Log Service provides data analysis, storage, and visualization features that are extensible.



References

- Free resources
- Limits on resources
- Log data format
- Instructions

14.2. Application operation and management

This topic describes the details of the COVID-19 Pandemic Analysis application, including the overview, assets, limits, data description, and usage notes.

Overview

> **Note** The COVID-19 Pandemic Analysis application is being upgraded. You cannot create related resources.

- The first time you use the application, you must initialize it. The initialization process takes approximately two minutes.
- Data is automatically updated every day.
- Data sources: Novel Coronavirus (COVID-19) Cases, provided by JHU CSSE.
- Update frequency: Once a day around 23:59 (UTC).
- Data is collected from multiple sources that update at different times and may not always align.
- You are not charged for data storage or data analysis.
- If you do not use the free resources of the application for a long time, the resources may be reclaimed. To continue using the free resources, you must re-initialize the application.

Assets

The following free resources are provided in the application:

- Project: ncp-{Alibaba Cloud account UID}-cn-chengdu
- Logstore: ncp
- Dashboards: covid-19_global and covid-19_detail.

Limits

- You cannot modify or delete the ncp Logstore, the indexes created for the Logstore, or the data written to the Logstore. Other than these limits, you can perform the same operations on the Logstore as on normal Logstores.
- You can create Logstores in the dedicated project that is automatically created in the application and write data to the Logstores. You are charged when you use the Logstores.
- Modifications to the dedicated dashboards are overwritten during updates. We recommend that you do not modify the dashboards. You can copy the dedicated dashboards to another project and modify the copies. For more information, see How do I copy an existing dashboard to create a dashboard?

Dashboards

The COVID-19 Pandemic Analysis application provides multiple built-in dashboards. The dashboards are created based on the log data in the ncp Logstore. You can also create custom dashboards.

Dashboard	ID	Description
COVID-19 Global	covid-19_global	Displays various metrics, trend charts, and the COVID-19 pandemic situation of each country and region around the world.
COVID-19 Detail	covid-19_detail	Displays various metrics, trend charts, and the COVID-19 pandemic situation of each province or state in all countries and regions around the world.

Data description

- Data updates and usage notes

  COVID-19 pandemic datasets are stored in the ncp Logstore. Data updates are automatically synchronized to the Logstore every day. Each update is represented by a value in the Version field, for example, v2020-01-26T12:30:00.

  Every update contains full data. You can query and analyze the most recent update to obtain the data that you want.

  You can specify a value for the Version field to query and analyze the data in the specified update, as shown in the following query statement.

  ```
  Version: "v2020-01-26T12:30:00" and Type : "Province/State Cases" | select .... from log
  ```

  We recommend that you execute the following query statement to query and analyze data. This ensures that you can query the most recent update.

```
Type : "Province/State Cases" | select .... from log l right join (select max(Version) as Version from log) r on  l.Version =  r.Version
```

> **Note**
>
> - The code that precedes | is a search statement. In most cases, you can specify a value for the Type field to filter specific types of log data. For more information about the search syntax, see Search syntax.
> - The code that follows | is an analytic statement that uses the SQL-92 syntax. The from log clause is used to query log data from the current Logstore. You can specify a JOIN clause to query data from multiple Logstores. In addition, you can specify a JOIN clause to query data from external data sources, such as IP geolocation databases and the external tables of Object Storage Service (OSS) or ApsaraDB RDS for MySQL. For more information, see SQL syntax and functions.
> - Data is automatically updated every day. You can set the time range to 1 Day(Relative) for your queries.

- Overview

  COVID-19 pandemic datasets are stored in the ncp Logstore and can be identified based on value of the Type field. The valid values of the Type field include Global Cases, Country/Region Cases, and Province/State Cases.

- Global Cases

> **Note** **Hist** data is displayed in mini-charts in the table. **Trend** data is displayed in trend charts.

Field	Description	Example
Type	Type	Valid value: Global Cases.
Version	The version of the data update.	v2020-01-26T12:30:00
Last Update	The time when the most recent update was released.	2020-01-26 18:23
Confirmed	The most recent total number of confirmed cases.	1058
Confirmed Hist	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is an array.	[270, 444, 444, 549, 729, 1058]
Confirmed Trend	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
Recovered	The most recent total number of recoveries.	42

Field	Description	Example
Recovered Hist	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is an array.	[0, 28, 28, 31, 32, 42]
Recovered Trend	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
Deaths	The most recent total number of deaths.	52
Deaths Hist	The total number of COVID-19 deaths since January 23, 2020. The value of this field is an array.	[3, 17, 17, 24, 39, 52]
Deaths Trend	The total number of COVID-19 deaths since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
New Confirmed Hist	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is an array.	[11, 0, 41, 0, 56, 127]
New Confirmed Trend	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 7}

- Country/Region Cases

> **Note** **Hist** data is displayed in mini-charts in the table. **Trend** data is displayed in trend charts.

Field	Description	Example
Type	Type	Valid value: Country/Region Cases.
Version	The version of the data update.	v2020-01-26T12:30:00
Last Update	The time when the most recent update was released.	2020-01-26 18:23
Country/Region	The name of the country or region.	China, US

Field	Description	Example
LatLng	The longitude and latitude of the country or region. The value is a string in the <latitude>, <longitude> format.	51.7283857,-2.2085499
Confirmed	The most recent total number of confirmed cases.	1058
Confirmed Hist	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is an array.	[270, 444, 444, 549, 729, 1058]
Confirmed Trend	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
Recovered	The most recent total number of recoveries.	42
Recovered Hist	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is an array.	[0, 28, 28, 31, 32, 42]
Recovered Trend	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
Deaths	The most recent total number of deaths.	52
Deaths Hist	The total number of COVID-19 deaths since January 23, 2020. The value of this field is an array.	[3, 17, 17, 24, 39, 52]
Deaths Trend	The total number of COVID-19 deaths since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
New Confirmed Hist	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is an array.	[11, 0, 41, 0, 56, 127]
New Confirmed Trend	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 7}

- Province/State Cases

> **Note**
> - If the source data does not specify the province or state information, the value of the Province/State field is Unspecified *.
> - **Hist** data is displayed in mini-charts in the table. **Trend** data is displayed in trend charts.

Field	Description	Example
Type	Type	Valid value: Province/State Cases.
Version	The version of the data update.	v2020-01-26T12:30:00
Last Update	The time when the most recent update was released.	2020-01-26 18:23
Country/Region	The name of the country or region.	China, US
Province/State	The name of the province or state.	Shanghai, New York
LatLng	The longitude and latitude of the country or region. The value is a string in the <latitude>,<longitude> format.	51.7283857,-2.2085499
Confirmed	The most recent total number of confirmed cases.	1058
Confirmed Hist	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is an array.	[270, 444, 444, 549, 729, 1058]
Confirmed Trend	The total number of confirmed COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
Recovered	The most recent total number of recoveries.	42
Recovered Hist	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is an array.	[0, 28, 28, 31, 32, 42]
Recovered Trend	The total number of COVID-19 recoveries since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}

Field	Description	Example
Deaths	The most recent total number of deaths.	52
Deaths Hist	The total number of COVID-19 deaths since January 23, 2020. The value of this field is an array.	[3, 17, 17, 24, 39, 52]
Deaths Trend	The total number of COVID-19 deaths since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 3}
New Confirmed Hist	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is an array.	[11, 0, 41, 0, 56, 127]
New Confirmed Trend	The total number of suspected COVID-19 cases since January 23, 2020. The value of this field is a dictionary.	{"2020-01-21": 1, "2020-01-22": 1, "2020-01-23": 1, "2020-01-24": 2, "2020-01-25": 2, "2020-01-26": 7}

Usage notes

1.
2. On the **Social Welfare** tab of the **Log Application** section, click **COVID-19 Epidemic Analysis**.
3. Use the COVID-19 Epidemic Analysis application as prompted.

FAQ

- How do I delete the dedicated project of COVID-19 Pandemic Analysis?

  If you want to delete the ncp-{Alibab Cloud account UID}-cn-chengdu project, run the following command in Cloud Shell:

  ```
  aliyunlog log delete_project --project_name=ncp-$ALICLOUD_ACCOUNT_ID-cn-chengdu --region-endpoint=cn-chengdu.log.aliyuncs.com
  ```

  Notice If you delete the dedicated project, the Logstores that you created in the project are also deleted.

- How do I copy an existing dashboard to create a dashboard?

  i. In the upper-right corner of the Alibaba Cloud console, click the Cloud Shell icon.

  ii. Copy the following configurations of the dashboard to the local host:

```
aliyunlog log get_dashboard --project=ncp-$ALICLOUD_ACCOUNT_ID-cn-chengdu --entity=co
vid-19_global --region-endpoint=cn-chengdu.log.aliyuncs.com > covid-19_global.json
aliyunlog log get_dashboard --project=ncp-$ALICLOUD_ACCOUNT_ID-cn-chengdu --entity=co
vid-19_detail --region-endpoint=cn-chengdu.log.aliyuncs.com > covid-19_detail.json
sed -i "s/\"dashboardName\": \"/\"dashboardName\": \"v2/g" covid-19_global.json
sed -i "s/\"description\": \"\", \"displayName\": \"/\"description\": \"\", \"display
Name\": \"v2/g" covid-19_global.json
sed -i "s/\"dashboardName\": \"/\"dashboardName\": \"v2/g" covid-19_detail.json
sed -i "s/\"description\": \"\", \"displayName\": \"/\"description\": \"\", \"display
Name\": \"v2/g" covid-19_detail.json
aliyunlog log create_dashboard --project=ncp-$ALICLOUD_ACCOUNT_ID-cn-chengdu --detail
=file://./covid-19_global.json --region-endpoint=cn-chengdu.log.aliyuncs.com
aliyunlog log create_dashboard --project=ncp-$ALICLOUD_ACCOUNT_ID-cn-chengdu --detail
=file://./covid-19_detail.json --region-endpoint=cn-chengdu.log.aliyuncs.com
```

iii. View the created dashboard.

In the left-side navigation pane of the **COVID-19 Pandemic Analysis** page, click **Setup**. On the page that appears, click **Go to Overview Page**. In left-side navigation pane of the page that appears, click **Dashboard** to view the created dashboard.

References

- What is Log Service?
- Visualization overview