ALIBABA CLOUD

Alibaba Cloud

Log Service

Data Collection

Document Version: 20220711

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.Data collection overview 12
2.Collection acceleration 15
2.1. Enable the global acceleration feature 15
2.2. Configure log collection acceleration for Logtail 16
2.3. Disable the global acceleration feature 16
3.Logtail collection 18
3.1. Overview 18
3.1.1. Use Logtail to collect data 18
3.1.2. Log collection process of Logtail 21
3.1.3. Logtail configuration files and record files 24
3.2. Select a network type 32
3.3. Install 35
3.3.1. Install Logtail on ECS instances 35
3.3.2. Install Logtail on a Linux server 39
3.3.3. Install Logtail on a Windows server 56
3.3.4. Configure the startup parameters of Logtail 61
3.4. Machine Group 75
3.4.1. Introduction 75
3.4.2. Configure a user identifier 76
3.4.3. Create an IP address-based machine group 78
3.4.4. Create a custom ID-based machine group 80
3.4.5. Manage machine groups 83
3.4.6. Manage Logtail configurations for log collection 84
3.5. Text logs 85
3.5.1. Overview 85
3.5.2. Collect logs in simple mode 86

3.5.3. Collect logs in full regex mode 94
3.5.4. Collect logs in NGINX mode 102
3.5.5. Collect logs in delimiter mode 110
3.5.6. Collect logs in JSON mode 118
3.5.7. Collect logs in IIS mode 125
3.5.8. Collect logs in Apache mode 133
3.5.9. Import historical logs 143
3.5.10. Time formats 146
3.5.11. Log topics 149
3.6. Container log collection 151
3.6.1. Overview 152
3.6.2. Install Logtail components 153
3.6.3. Use the Log Service console to collect container text lo.. 159
3.6.4. Use the Log Service console to collect container stdout... 174
3.6.5. Use CRDs to collect container logs in DaemonSet mode 193
3.6.6. Use CRDs to collect container text logs in Sidecar mod... 207
3.6.7. Use the Log Service console to collect container text lo... 221
3.6.8. Collect logs from standard Docker containers 226
3.6.9. Collect Kubernetes events 231
3.7. Customize Logtail plug-ins to collect data 235
3.7.1. Overview 235
3.7.2. Collect MySQL binary logs 236
3.7.3. Collect MySQL query results 248
3.7.4. Collect HTTP data 254
3.7.5. Collect syslogs 258
3.7.6. Collect data from Beats and Logstash 263
3.7.7. Collect systemd-journald logs 267
3.7.8. Collect Docker events 273

3.7.9. Collect Windows event logs 275
3.8. Customize Logtail plug-ins to process data 282
3.8.1. Overview 283
3.8.2. Extract fields 285
3.8.3. Add fields 294
3.8.4. Drop fields 295
3.8.5. Rename columns 296
3.8.6. Encapsulate fields 297
3.8.7. Expand JSON fields 298
3.8.8. Filter logs 300
3.8.9. Extract log time 302
3.8.10. Transform IP addresses 308
3.8.11. Append data to a field 310
3.9. Use built-in alert monitoring rules for Logtail 312
3.10. Logtail limits 317
3.11. FAQ about Logtail 321
4.Cloud product collection 323
4.1. Alibaba Cloud service logs 323
4.2. Common operations on logs of Alibaba Cloud services 329
4.3. Function Compute execution logs 330
4.3.1. Overview 330
4.3.2. Enable the logging feature 331
4.3.3. Log fields 332
4.4. OSS access logs 332
4.4.1. Usage notes 332
4.4.2. Enable the real-time log query feature 333
4.4.3. Log fields 334
4.5. NAS access logs 343

4.5.1. Usage notes 343
4.5.2. Enable the log analysis feature 344
4.5.3. Log fields 345
4.6. Anti-DDoS Pro logs 346
4.6.1. Usage notes 346
4.6.2. Enable the full log feature 347
4.6.3. Log fields 348
4.7. Anti-DDoS Pro and Anti-DDoS Premium logs 350
4.7.1. Usage notes 350
4.7.2. Enable the log analysis feature 352
4.7.3. Manage log storage 354
4.7.4. Log fields 355
4.8. Anti-DDoS Origin logs 357
4.8.1. Usage notes 357
4.8.2. Enable the mitigation analysis feature of Anti-DDoS Or... 358
4.8.3. Log fields 359
4.9. Security Center logs 361
4.9.1. Usage notes 361
4.9.2. Enable the log analysis feature 362
4.9.3. Log fields 363
4.10. WAF logs 375
4.10.1. Usage notes 375
4.10.2. Enable the log analysis feature 377
4.10.3. Manage the log storage space 378
4.10.4. Log fields 378
4.11. Cloud Firewall logs 388
4.11.1. Usage notes 388
4.11.2. Enable the log analysis feature 389

4.11.3. Manage log storage space 390
4.11.4. Log fields 390
4.12. Layer 7 access logs for SLB 392
4.12.1. Usage notes 392
4.12.2. Enable the access log management feature 394
4.12.3. Log fields 395
4.13. Layer 4 monitoring metrics for SLB 396
4.13.1. Usage notes 396
4.13.2. Enable the Layer 4 monitoring feature 397
4.13.3. Layer 4 monitoring metrics 398
4.14. VPC flow logs 399
4.14.1. Usage notes 399
4.14.2. Enable the flow log feature 402
4.14.3. Log fields 404
4.15. EIP logs 405
4.15.1. Overview 405
4.15.2. Enable fine-grained monitoring 406
4.15.3. Log fields 407
4.16. API Gateway access logs 408
4.16.1. Usage notes 408
4.16.2. Enable the log management feature 410
4.16.3. Log fields 410
4.17. ActionTrail access logs 411
4.17.1. Usage notes 411
4.17.2. Enable the log shipping feature 412
4.17.3. Log fields 413
4.18. Inner-ActionTrail 415
4.18.1. Usage notes 415

4.18.2. Enable the inner-ActionTrail feature 416
4.18.3. Fields in Alibaba Cloud-initiated events 418
4.19. PolarDB-X 1.0 SQL audit logs 420
4.19.1. Usage notes 420
4.19.2. Enable the SQL audit and analysis feature 421
4.19.3. Log fields 423
4.20. RDS SQL execution logs 424
4.20.1. Usage notes 424
4.20.2. Collect RDS SQL audit logs 426
4.20.3. Log fields 427
4.21. ApsaraDB for Redis logs 428
4.21.1. Usage notes 428
4.21.2. Enable the log audit feature 429
4.21.3. Log fields 430
4.22. MongoDB logs 432
4.22.1. Usage notes 432
4.22.2. Enable the log audit feature 433
4.22.3. Log fields 434
4.23. IoT Platform logs 436
4.23.1. Usage notes 437
4.23.2. Enable the log dump feature 437
4.23.3. Log fields 438
4.24. DCDN real-time logs 440
4.24.1. Usage notes 440
4.24.2. Enable the real-time log delivery feature 442
4.24.3. Log fields 444
5.Data import 447
5.1. Import data from OSS to Log Service 447

5.2. Import data from MaxCompute to Log Service 455
5.3. Time formats 458
6.Other collection methods 461
6.1. Use web tracking to collect logs 461
6.2. Use the Kafka protocol to upload logs 464
6.3. Use the syslog protocol to upload logs 472
6.4. Logstash 477
6.4.1. Install Logstash 477
6.4.2. Create Logstash configurations for log collection and p... 479
6.4.3. Configure Logstash as a Windows service 483
6.4.4. Advanced features 484
6.4.5. Logstash error handling 484
6.5. Use SDKs to collect logs 484
7.Collect common logs 487
7.1. Collect Log4j logs 487
7.2. Collect Python logs 492
7.3. Collect Node.js logs 495
7.4. Collect WordPress logs 497
7.5. Collect Unity3D logs 503
8.Best practices 506
8.1. Collect IoT or embedded development logs 506
8.2. Use web tracking to collect logs 514
8.3. Build a service to upload logs from mobile apps to Log S... 520
8.4. Collect Zabbix data 524
8.5. Collect logs across Alibaba Cloud accounts 526
8.6. Collect container logs across Alibaba Cloud accounts 531
9.FAQ 536
9.1. FAQ about data collection 536

9.2. Log management 536
9.3. FAQ about Logtail 537
9.4. How do I collect logs from servers in a corporate intranet? 539
9.5. Troubleshoot log collection exceptions in containers 542
9.6. How do I obtain the labels and environment variables of ... 545
9.7. Query local collection status 547
9.8. What do I do if errors occur when I use Logtail to collect... 561
9.9. What do I do if a Logtail machine group has no heartbea... 562
9.10. How do I use the automatic diagnostic tool of Logtail? 567
9.11. How do I update a Logtail configuration after I switch th.. 574
9.12. How do I view Logtail collection errors? 574
9.13. How do I troubleshoot the common errors that occur wh... 576
9.14. How do I optimize regular expressions? 588
9.15. How do I collect different types of logs in full regex mod.. 588
9.16. Why am I unable to collect SLB access logs? 589
9.17. What are the differences among log collection agents? 590
9.18. What are the differences between LogHub and Kafka? 595
9.19. What do I do if I want to use multiple Logtail configurat... 597

1.Data collection overview

Log Service allows you to collect data from multiple sources, such as servers, applications, open source software, IoT devices, mobile devices, and Alibaba Cloud services. You can also collect data that is transferred over standard protocols. This topic describes the data sources that are supported by Log Service.

Data sources

The following table describes the sources from which Log Service can collect data.

Category	Source	Collection method	References
Application	Program output	Logtail	None
	Access log	Logtail	Collect and analyze NGINX access logs
Programming language	Java	• Log Service Java SDK • Java Producer Library	None
	Log4J Appender	• 1.x • 2.x	None
	LogBack Appender	LogBack	None
	C	Log Service C SDK	None
	Python	Log Service Python SDK	None
	Python Logging	None	None
	PHP	Log Service PHP SDK	None
	.NET	Log Service csharp SDK	None
	C++	Log Service C++ SDK	None
	Go	• Log Service Go SDK • Golang Producer Library	None
	Node.js	NodeJs	None
	JS	JS/Web Tracking	None
	Linux	Logtail	None
	Windows	Logtail	None
	MAC/Unix	Native C	None

Category	Source	Collection method	References
OS	Docker file	Use Logtail to collect Docker files	None
	Docker output	Use Logtail to collect container logs	None
Database	MySQL binary log	Collect MySQL binary logs	None
	JDBC SELECT	Collect MySQL query results	None
Mobile device	iOS and Android	• Log Service Android SDK • Log Service iOS SDK	None
	Web page	JS/Web Tracking	None
	Intelligent IoT	C Producer Library	None
Standard protocol	HTTP polling data	Logtail HTTP	Collect and analyze NGINX monitoring logs
	Syslog	Use Logtail to collect syslog logs	None
Imported data	MaxCompute	Import data from MaxCompute to Log Service	None
	Object Storage Service (OSS)	Import data from OSS to Log Service	None
	Flink	Use Flink to write data to Log Service	Register a Log Service project
Third-party software	Logstash	Logstash and Use the Kafka protocol to upload logs	None
	Flume	Use Flume to consume log data	None
	Beats	Use the Kafka protocol to upload logs	None
	Fluentd	Use the Kafka protocol to upload logs	None
	Telegraf	Use the Kafka protocol to upload logs	None

Category	Source	Collection method	References
Alibaba Cloud service	Services such as Elastic Compute Service (ECS) and OSS	Alibaba Cloud service logs	None

Endpoints

Log Service provides the following types of endpoints:

- Internal endpoint: If you select an internal endpoint, Log Service can be accessed by other Alibaba Cloud services in the same region over the classic network or a virtual private cloud (VPC), which is reliable.
- Public endpoint: If you select a public endpoint, Log Service can be accessed over the Internet, and the data transmission speed varies based on network conditions. We recommend that you use HTTPS to ensure secure data transmission.

For more information, see Endpoints.

FAQ

- Which network type do I need to select if I use Express Connect together with Log Service?

  We recommend that you select the internal network type.

- Can I obtain public IP addresses when I collect data from the Internet?

  Yes, you can turn on Log Public IP for your Logstore to obtain public IP addresses. For more information, see Create a Logstore.

- Which network type do I need to select if I want to collect ECS logs from Region A to a Log Service project in Region B?

  We recommend that you select the Internet type and specify the region name of Region B in the command used to install Logtail on your ECS instance in Region A. This way, the Logtail can access the project in Region B over the Internet. For more information about how to select a network type in other scenarios, see Select a network type.

- How do I determine whether an endpoint is accessible?

  You can run the following command. If output is returned, the endpoint is accessible.

  ```
  curl $myproject.cn-hangzhou.log.aliyuncs.com
  ```

  - `$myproject`: the project name
  - `cn-hangzhou.log.aliyuncs.com`: the endpoint

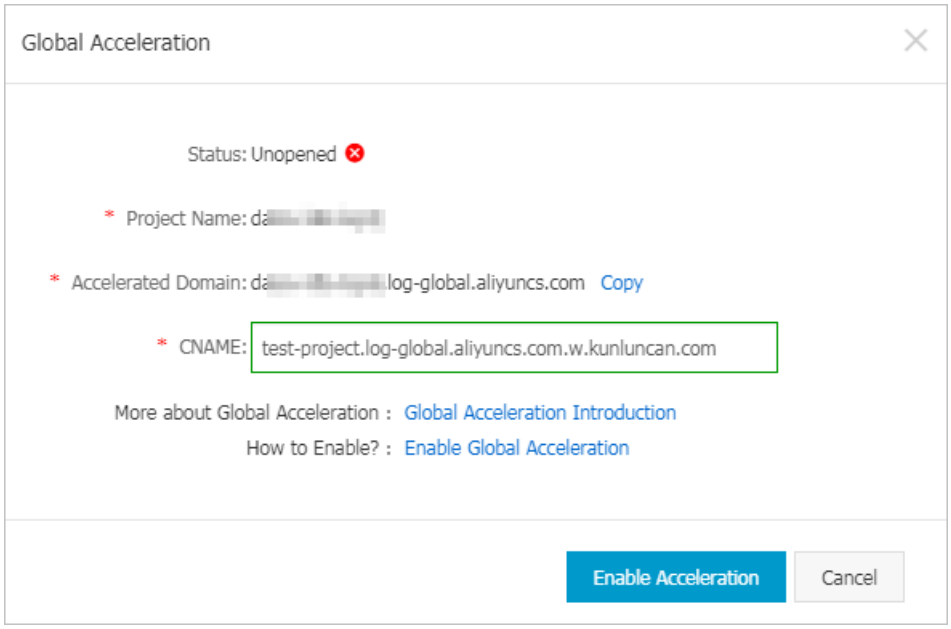
2.Collection acceleration

2.1. Enable the global acceleration feature

This topic describes how to enable the global acceleration feature of Log Service in Log Service console.

Procedure

1. Submit a ticket to obtain a canonical name (CNAME).
2. Enable the global acceleration feature.
   i. Log on to the Log Service console.
   ii. In the project list, click the project for which you want to enable the global acceleration feature.
   iii. On the **Overview** page, click **Modify** next to **Global Acceleration**.
   iv. In the **Global Acceleration** panel, enter the **CNAME**, and then click **Enable Acceleration**.



What to do next

After you enable the global acceleration feature, you can configure collection acceleration.

- Use Logtail to collect logs.
  - If you enable the global acceleration feature before you install Logtail, select **Global Acceleration** when you install Logtail. For more information, see Install Logtail on a Linux server.
  - If you enable the global acceleration feature after you install Logtail, switch the collection mode to **Global Acceleration**. For more information, see Configure log collection acceleration for Logtail.
- Use an SDK to collect logs.

If you use an SDK to collect logs, you can replace the specified endpoint of the Log Service project with `log-global.aliyuncs.com`.

2.2. Configure log collection acceleration for Logtail

This topic describes how to switch the collection mode of Logtail to Global Acceleration after you install Logtail.

Prerequisites

The global acceleration feature is enabled. For more information, see Enable the global acceleration feature.

Configurations

- If you enable the global acceleration feature before you install Logtail, select **Global Acceleration** when you install Logtail. For more information, see Install Logtail on a Linux server.
- If you enable the global acceleration feature after you install Logtail, follow the instructions in this topic to switch the collection mode of Logtail to **Global Acceleration**.

Procedure

1. Stop Logtail.
   - Linux:

     Run the `/etc/init.d/ilogtaild stop` command as the root user.
   - Windows:
     a. Choose **Start Menu > Control Panel > Administrative Tools > Services**.
     b. On the **Services** window, right-click the LogtailWorker service, and select **Stop**.
2. Modify the Logtail startup configuration file *ilogtail_config.json*.

   Replace the endpoint in the data_server_list parameter with `log-global.aliyuncs.com`. For more information, see Startup configuration file (ilogtail_config.json).
3. Start Logtail.
   - Linux:

     Run the `/etc/init.d/ilogtaild start` command as the root user.
   - Windows:
     a. Choose **Start Menu > Control Panel > Administrative Tools > Services**.
     b. On the **Services** window, right-click the LogtailWorker service, and select **Start**.

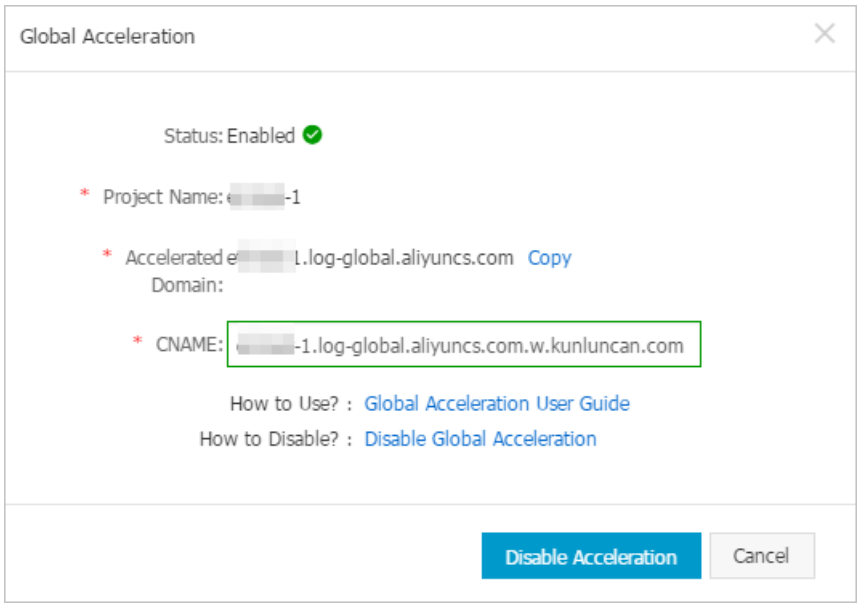
2.3. Disable the global acceleration feature

This topic describes how to disable the global acceleration feature of Log Service in Log Service console.

Procedure

1.
2.
3. On the **Overview** page, click **Modify** next to **Global Acceleration**.
4. In the **Global Acceleration** panel, enter the **CNAME**, and click **Disable Acceleration**.

> **Notice** Before you disable the global acceleration feature, make sure that you no longer use this domain name to upload or request data.



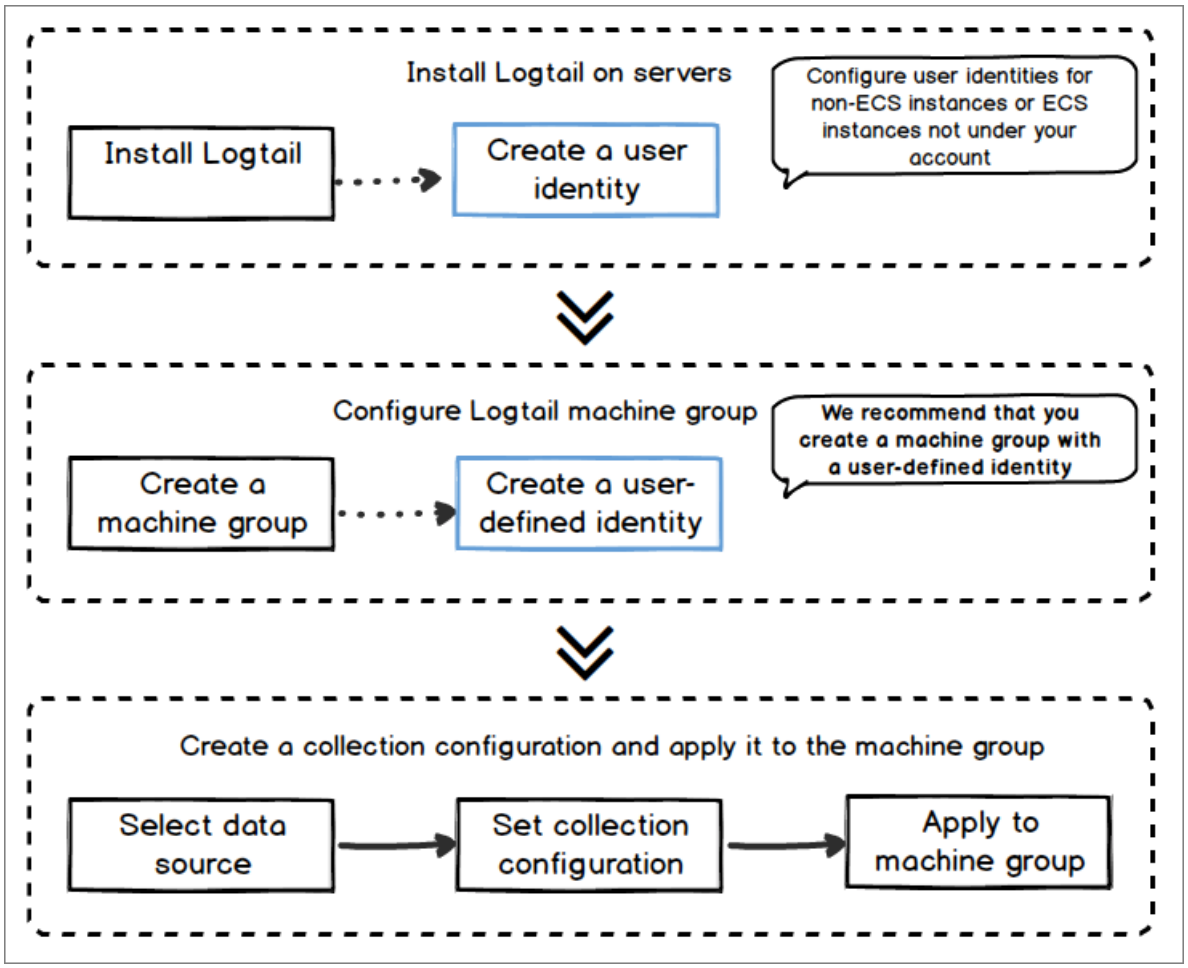
3.Logtail collection

3.1. Overview

3.1.1. Use Logtail to collect data

Logtail is a log collection agent that is provided by Log Service. You can use Logtail to collect logs from multiple data sources in real time. These sources include Alibaba Cloud Elastic Compute Service (ECS) instances, data centers, and servers that belong to third-party cloud service providers. This topic describes the features, benefits, limits, and configuration process of Logtail.

Configuration process



1. Install Logtail on a server.
   - For more information about how to install Logtail on Linux, see Install Logtail on a Linux server.
   - For more information about how to install Logtail on Windows, see Install Logtail on a Windows server.
2. The server on which you want to install Logtail can be an ECS instance that does not belong to the current account, or belongs to a local data center or third-party cloud service provider. In this case, you must specify a user identity for the server.

   For more information, see Configure a user identifier.

3. Create a machine group.
   - For more information about how create an IP address-based machine group, see Create an IP address-based machine group.
   - For more information about how to create a custom identifier-based machine group, see Create a custom ID-based machine group.
4. Create Logtail configurations and apply the configurations to the machine group.

   You can perform the preceding operations in the Log Service console. For more information, see Overview.

After you perform the preceding operations, Logtail collect logs from you server and send the logs to the specified Logstore. You can query logs by using the Log Service console, API operations, SDKs, or CLI.

Benefits

- Supports non-intrusive log collection based on log files. You do not need to modify your application code. Your applications are not affected when Logtail collects logs.
- Allows you to collect text logs, binary logs, HTTP data, and container logs.
- Allows you to collect logs from standard containers, swarm clusters, and Kubernetes clusters.
  - For more information about how to collect logs from swarm clusters, see Enable Log Service.
  - For more information about how to collect logs from Container Service for Kubernetes, see Overview.
  - For more information about how to collect logs from self-managed Kubernetes clusters, see Overview.
  - For more information about how to collect logs from self-managed Docker clusters, see Collect logs from standard Docker containers.
- Handles exceptions during log collection. If a network or server exception occurs, Logtail retries log collection and caches logs on local servers to ensure data security.
- Provides centralized management based on Log Service. After you install Logtail on servers and create a machine group and Logtail configurations, Logtail collects logs from the servers and sends the logs to Log Service.
- Provides a comprehensive self-protection mechanism. The CPU, memory, and network resources that Logtail can use are limited. This ensures that Logtail does not affect the performance of other services on the server.

Limits

For more information about the limits of Logtail, see Logtail limits.

Terms

- Machine group: A machine group contains one or more servers from which logs of a specific type are collected. After you apply Logtail configurations to a machine group, Log Service collects logs from the servers in the machine group based on the configurations.

  You can set an IP address-based identifier or a custom identifier for a machine group. Then, you can manage the servers in the machine group based on the identifier. You can create and delete a machine group, add servers to a machine group, and remove servers from a machine group in the Log Service console.
- Logtail is a log collection agent that is provided by Log Service. Logtail runs on servers from which

you want to collect logs.

- For a Linux-based server, Logtail is installed in the */usr/local/ilogtail* directory. Logtail initiates the following separate processes whose names start with ilogtail: a log collection process and a daemon process. The logs of Logtail are stored in the */usr/local/ilogtail/ilogtail.LOG* directory.
- For a Windows-based server, Logtail is installed in the *C:\Program Files\Alibaba\Logtail* directory (32-bit system) or *C:\Program Files (x86)\Alibaba\Logtail* directory (64-bit system). Choose **Control Panel > Administrative Tools > Services**. On the Services window, you can view the LogtailDaemon service. The logs of Logtail are stored in the `ilogtail.LOG` file.

- Logtail configurations for log collection: Logtail configurations for log collection are a set of policies that Logtail uses to collect logs. You can specify the data source and collection mode to create custom Logtail configurations for log collection. The configurations specify how to collect logs from servers, parse the logs, and send the logs to a specified Logstore.

Features

Feature	Description
Log collection in real time	Logtail monitors log files, and reads and parses incremental logs in real time. In most cases, logs are sent to Log Service within 3 seconds after the logs are generated. For more information, see Log collection process of Logtail. **Note** Logtail does not collect historical data. If a log entry is generated for more than 12 hours before the log is read, Logtail does not collect the log entry.
Automatic log rotation	Multiple applications rotate log files based on the file size or date. The original log file is renamed and an empty log file is created during the rotation process. For example, the *app.LOG* file is renamed *app.LOG.1* and *app.LOG.2* during log rotation. You can specify the file to which collected logs are written, for example, *app.LOG*. Logtail monitors the log rotation process to ensure that no logs are lost.
Multiple data sources	Logtail can collect text logs, syslogs, HTTP logs, and MySQL binlogs. For more information, see Data collection overview.
Compatibility with open source collection agents	You can use open source agents such as Logstash and Beats to collect data. Then, you can use Logtail to collect data from the agents and send the data to Log Service. For more information, see Data collection overview.
Automatically handle collection exceptions	If data fails to be sent to Log Service due to exceptions, Logtail retries to collect logs based on the scenario. The exceptions include server errors, network errors, and quota exhaustion. If the retry fails, Logtail writes the data to the local cache and resends the data after 3 seconds. For more information, see How do I use the automatic diagnostic tool of Logtail?.

Feature	Description
Flexible configurations	Logtail allows you to create configurations for log collection in a flexible manner. You can specify the directories and files from which logs are collected. You can also specify an exact match or a wildcard match based on your business requirements. You can also specify the log collection mode and customize the fields that you want to extract. You can use a regular expression to extract fields from logs. Log data in Log Service must have the timestamp information. Logtail allows you to customize log time formats and then extract the required timestamps from the time information based on different formats.
Automatically synchronize Logtail configurations	After you create or update Logtail configurations for log collection in the Log Service console, the configurations are synchronized to the servers in which Logtail is installed and take effect within 3 minutes. Logs are collected based on the original configurations during the synchronization.
Status monitoring	Logtail monitors the CPU and memory resources that are consumed in real time. This ensures that Logtail does not consume an excessive number of resources or affect other services. If the resource consumption exceeds the limit, Logtail is automatically restarted. Logtail also monitors the network bandwidth resources that are consumed. This ensures that Logtail does not consume an excessive amount of bandwidth. For more information, see Startup configuration file (ilogtail_config.json).
Signature and encryption	Logtail retrieves the AccessKey pair of your Alibaba Cloud account and uses the pair to sign all log data that is sent to Log Service. This way, data tampering is prevented during data transmission. **Note** Logtail retrieves the AccessKey pair of your Alibaba Cloud account by using the HTTPS protocol to ensure the security of your AccessKey pair.

Data collection reliability

Logtail stores checkpoints that are periodically collected to the local server during log collection. If an exception such as an unexpected server shutdown or a process failure occurs, Logtail restarts and then collects data from the last checkpoint. This process avoids incomplete data collection. Logtail runs based on the startup parameters that are specified in the startup configuration file. If the usage of a resource exceeds the limit for more than 5 minutes, Logtail is forcibly restarted. After the restart, a small amount of duplicate data may be collected to the specified Logstore.

To improve log collection reliability, Logtail uses multiple internal mechanisms. However, logs may fail to be collected in the following scenarios:

- Logtail is not running, but logs are rotated multiple times.
- The log rotation rate is high, for example, one rotation per second.
- The log collection rate is lower than the log generation rate for a long period of time.

3.1.2. Log collection process of Logtail

This topic describes how Logtail collects logs. The log collection process consists of the following steps: monitor log files, read log files, process logs, filter logs, aggregate logs, and sends logs.

Procedure

Logtail performs the following steps to collect log data:

1. Monitor log files
2. Read log files
3. Process logs
4. Filter logs
5. Aggregate logs
6. Send logs

Monitor log files

After you install Logtail on servers and create a Logtail configuration in the Log Service console, the configuration is synchronized to the servers in real time. Logtail monitors log files of the servers based on the configuration. Logtail scans log directories and files based on the log file path and the maximum directory depth that you specify for monitoring in the configuration.

If the log files of the servers in a machine group are not updated after the Logtail configuration is applied to the machine group, the log files are considered historical log files. Logtail does not collect historical log files. If log files are updated, Logtail reads and collects the files, and then sends the log files to Log Service. For more information about how to collect historical log files, see Import historical logs.

Logtail registers event listeners to monitor directories from which log files are collected. The event listeners poll the log files in the directories on a regular basis. This ensures that logs are collected at the earliest opportunity in a stable manner. For Linux servers, Inotify is used to monitor directories and poll log files.

Read log files

After Logtail detects updated log files, Logtail reads the log files.

- The first time Logtail reads a log file, Logtail can read up to 1,024 KB of data in the log file by default.
  - If the file size is less than 1,024 KB, Logtail reads data from the beginning of the file.
  - If the file size is greater than 1,024 KB, Logtail reads the last 1,024 KB of data in the file.

  > **Note** Log Service allows you to specify the data size that Logtail can read from a log file the first time Logtail reads the file.
  >
  > - Console mode: Modify the **First Collection Size** parameter in the Advanced Options section on the Logtail Config page. For more information, see Advanced Options.
  > - API mode: Modify the **tail_size_kb** parameter in the Logtail configuration. For more information, see advanced.

- If a log file is read before, Logtail reads the file from the previous checkpoint.
- Logtail can read up to 512 KB of data at a time. Make sure that the size of each log in a log file does not exceed 512 KB.

> **Note** If you change the system time on the server, you must restart Logtail. Otherwise, the log time becomes incorrect and logs are dropped.

Process logs

When Logtail reads a log file, Logtail splits each log in the file into multiple lines, parses the log, and then configures the time field for the log.

- Split a log into multiple lines

  If you specify a regular expression to match the first line of a log, Logtail splits the log into multiple lines based on the regular expression. If you do not specify a regular expression, a single log line is processed as a log.

- Parse logs

  Logtail parses each log based on the collection mode that you specify in the Logtail configuration.

  > **Note** If you specify complex regular expressions, Logtail may consume an excessive amount of CPU resources. We recommend that you specify regular expressions that allow Logtail to parse logs in an efficient manner.

  If Logtail fails to parse a log, Logtail handles the failure based on the setting of the Drop Failed to Parse Logs parameter in the Logtail configuration.

  - If you turn on **Drop Failed to Parse Logs**, Logtail drops the log and reports an error.
  - If you turn off **Drop Failed to Parse Logs**, Logtail uploads the log. The key of the log is set to **raw_log** and the value is set to the log content.

- Configure the time field for a log
  - If you do not configure the time field for a log, the log time is the time when the log is parsed.
  - If you configure the time field for a log, the manner in which the log is processed varies in the following scenarios:
    - If the difference between the time when the log is generated and the current time is within 12 hours, the log time is extracted from the parsed log fields.
    - If the difference between the time when the log is generated and the current time is greater than 12 hours, the log is dropped and an error is reported.

Filter logs

After logs are processed, Logtail filters the logs based on the specified filter conditions.

- If you do not specify filter conditions in the **Filter Configuration** field, the logs are not filtered.
- If you specify filter conditions in the **Filter Configuration** field, the fields in each log are traversed.

  Logtail collects only the logs that meet the filter conditions.

Aggregate logs

To reduce the number of network requests, Logtail caches the processed and filtered logs for a specified period of time. Then, Logtail aggregates the logs and sends the logs to Log Service. If one of the following conditions is met when data is cached, Logtail sends aggregated logs to Log Service.

- The aggregation duration exceeds 3 seconds.

- The number of aggregated logs exceeds 4,096.
- The total size of aggregated logs exceeds 512 KB.

Send logs

Logtail sends aggregated logs to Log Service. If a log fails to be sent, Logtail retries or no longer sends the log based on the HTTP status code.

HTTP status code	Description	Handling method
401	Logtail is not authorized to collect data.	Logtail drops the log packets.
404	The project or Logstore that is specified in the Logtail configuration does not exist.	Logtail drops the log packets.
403	The shard quota is exhausted.	Logtail tries again 3 seconds later.
500	A server exception occurs.	Logtail tries again 3 seconds later.

Note If you want to change the data transmission rate and the maximum number of concurrent connections, you can modify the max_bytes_per_sec and send_request_concurrency parameters in the Logtail startup configuration file. For more information, see Configure the startup parameters of Logtail.

3.1.3. Logtail configuration files and record files

This topic describes the basic configuration files and record files of Logtail. When Logtail is active, it uses the configuration files and generates record files.

Startup configuration file (ilogtail_config.json)

The *ilogtail_config.json* file is used to set Logtail startup parameters. For more information, see Configure the startup parameters of Logtail.

Note

- The file must be a valid JSON file. Otherwise, Logtail cannot be started.
- If you modify the file, you must restart Logtail to validate your modifications.

After Logtail is installed on a server, you can perform the following operations on the *ilogtail_config.json* file:

- Modify the Logtail runtime parameters.
- Check whether the command to install Logtail is correct.

  The values of the *config_server_address* and `data_server_list` parameters in the `ilogtail_config.json` file depend on the command that you select. If the region in the command is inconsistent with the region where the Log Service project is located or the URL in the command is inaccessible, the command is incorrect. In this case, Logtail cannot collect logs and must be reinstalled.
- File path

- In a Linux-based server: The file is stored in the */usr/local/ilogtail/ilogtail_config.json* directory.
- In a Windows-based server:
  - 64-bit: The file is stored in the *C:\Program Files (x86)\Alibaba\Logtail\ilogtail_config.json* directory.
  - 32-bit: The file is stored in the *C:\Program Files\Alibaba\Logtail\ilogtail_config.json* directory.
- Containers: The file is stored in a Logtail container. The file path is specified in the environment variable `ALIYUN_LOGTAIL_USER_ID` of the Logtail container. You can run the command `docker inspect ${logtail_container_name} | grep ALIYUN_LOGTAIL_USER_ID` to view the file path. Example: */etc/ilogtail/conf/cn_hangzhou/ilogtail_config.json*.

- Sample file

```
$cat /usr/local/ilogtail/ilogtail_config.json
{
    "config_server_address" : "http://logtail.cn-hangzhou-intranet.log.aliyuncs.com",
    "data_server_list" :
    [
        {
            "cluster" : "cn-hangzhou",
            "endpoint" : "cn-hangzhou-intranet.log.aliyuncs.com"
        }
    ],
    "cpu_usage_limit" : 0.4,
    "mem_usage_limit" : 100,
    "max_bytes_per_sec" : 2097152,
    "process_thread_count" : 1,
    "send_request_concurrency" : 4,
    "streamlog_open" : false
}
```

Configuration file of user identity

The configuration file of user identity contains the ID of your Alibaba Cloud account. The file is used to indicate that the Alibaba Cloud account is authorized to access the server and collect logs. For more information, see Configure a user identifier.

> **Note**
>
> - You must configure the user identity file when you collect logs from ECS instances that do not belong to your current Alibaba Cloud account, servers in your on-premises, or servers that are provided by third-party cloud service providers.
> - The configuration file of user identity must contain the ID of your Alibaba Cloud account. You cannot configure the ID of a RAM user for the file.
> - You need to configure the name of the user identity file. You do not need to configure the file extension.
> - You can configure multiple user identity files for a server. However, you can configure only one user identity file for a Logtail container.

- File path
  - In a Linux-based server: The file is stored in the */etc/ilogtail/users/* directory.

- In a Windows-based server: The file is stored in the `C:\LogtailData\users\` directory.
- Containers: The file is stored in a Logtail container. The file path is specified in the environment variable `ALIYUN_LOGTAIL_USER_ID` of the Logtail container. You can run the command `docker inspect ${logtail_container_name} | grep ALIYUN_LOGTAIL_USER_ID` to view the file path.

- Sample file

```
$ls /etc/ilogtail/users/
*************** *****************
```

Custom identifier file (user_defined_id)

The *user_defined_id* file is used to configure a custom ID for a machine group. For more information, see Create a custom ID-based machine group.

> **Note** If you want to configure a custom identifier for a machine group, you must configure the *user_defined_id* file.

- File path
  - In a Linux-based server: The file is stored in the */etc/ilogtail/user_defined_id* directory.
  - In a Windows-based server: The file is stored in the *C:\LogtailData\user_defined_id* directory.
  - Containers: The file is stored in a Logtail container. The file path is specified in the environment variable `ALIYUN_LOGTAIL_USER_DEFINED_ID` of the Logtail container. You can run the command `docker inspect ${logtail_container_name} | grep ALIYUN_LOGTAIL_USER_DEFINED_ID` to view the file path.

- Sample file

```
$cat /etc/ilogtail/user_defined_id
aliyun-ecs-rs1e16355
```

Logtail configuration file for log collection (user_log_config.json)

The *user_log_config.json* file records the Logtail configurations for log collection. The configurations are synchronized from Log Service. The file is updated whenever the Logtail configurations for log collection are updated. The file type is JSON. You can use the *user_log_config.json* file to check whether Logtail configurations for log collection are synchronized to the server. If the file exists and the configurations in the file are the same as the log collection configurations in the Log Service console, the synchronization succeeds.

> **Note** We recommend that you do not modify the file unless you need to specify sensitive information, such as the AccessKey pair and database password.

- File path
  - In a Linux-based server: The file is stored in the */usr/local/ilogtail/user_log_config.json* directory.
  - Windows
    - 64-bit: The file is stored in the *C:\Program Files (x86)\Alibaba\Logtail\user_log_config.json* directory.
    - 32-bit: The file is stored in the *C:\Program Files\Alibaba\Logtail\user_log_config.json* directory.

- Containers: The file is stored in the */usr/local/ilogtail/user_log_config.json* directory.

- Sample file

```
$cat /usr/local/ilogtail/user_log_config.json
{
   "metrics" : {
      "##1.0##k8s-log-c12ba2028*****939f0b$app-java" : {
         "aliuid" : "16542189*****50",
         "category" : "app-java",
         "create_time" : 1534739165,
         "defaultEndpoint" : "cn-hangzhou-intranet.log.aliyuncs.com",
         "delay_alarm_bytes" : 0,
         "enable" : true,
         "enable_tag" : true,
         "filter_keys" : [],
         "filter_regs" : [],
         "group_topic" : "",
         "local_storage" : true,
         "log_type" : "plugin",
         "log_tz" : "",
         "max_send_rate" : -1,
         "merge_type" : "topic",
         "plugin" : {
            "inputs" : [
               {
                  "detail" : {
                     "IncludeEnv" : {
                        "aliyun_logs_app-java" : "stdout"
                     },
                     "IncludeLable" : {
                        "io.kubernetes.container.name" : "java-log-demo-2",
                        "io.kubernetes.pod.namespace" : "default"
                     },
                     "Stderr" : true,
                     "Stdout" : true
                  },
                  "type" : "service_docker_stdout"
               }
            ]
         },
         "priority" : 0,
         "project_name" : "k8s-log-c12ba2028c*****ac1286939f0b",
         "raw_log" : false,
         "region" : "cn-hangzhou",
         "send_rate_expire" : 0,
         "sensitive_keys" : [],
         "tz_adjust" : false,
         "version" : 1
      }
   }
}
```

AppInfo record file (app_info.json)

The *app_info.json* file records the information about Logtail, such as the startup time, retrieved IP address, and hostname.

If you associate the IP address of the server with the hostname in the */etc/hosts* file, Logtail automatically retrieves the IP address of the server. If you do not associate the IP address of the server with the hostname, Logtail retrieves the IP address of the first network interface controller (NIC) on the server.

> **Note**
> - The AppInfo record file records only the information about Logtail.
> - If you modify the hostname of the server or other network configurations, you must restart Logtail to retrieve a new IP address.

- File path
  - In a Linux-based server: The file is stored in the */usr/local/ilogtail/app_info.json* directory.
  - Containers: The file is stored in the */usr/local/ilogtail/app_info.json* directory.
  - Windows
    - 64-bit: The file is stored in the *C:\Program Files (x86)\Alibaba\Logtail\app_info.json* directory.
    - 32-bit: The file is stored in the *C:\Program Files\Alibaba\Logtail\app_info.json* directory.
- Sample file

```
$cat /usr/local/ilogtail/app_info.json
{
   "UUID" : "",
   "hostname" : "logtail-ds-slpn8",
   "instance_id" : "E5F93BC6-B024-11E8-8831-0A58AC14039E_1**. ***. ***.***_1536053315",
   "ip" : "1**. ***. ***.***",
   "logtail_version" : "0.16.13",
   "os" : "Linux; 3.10.0-693.2.2.el7.x86_64; #1 SMP Tue Sep 12 22:26:13 UTC 2017; x86_64"
,
   "update_time" : "2018-09-04 09:28:36"
}
```

Field	Description
UUID	The serial number of the server.
hostname	The hostname of the server.
instance_id	The unique identifier of Logtail. This identifier is randomly generated.

Field	Description
ip	The IP address that is retrieved by Logtail. If Logtail does not retrieve an IP address, the value of this field is null. Logtail cannot run as expected. You must specify an IP address for the server and then restart Logtail. **Note** If you configure an IP address-based identifier for a machine group, make sure that the IP address-based identifier includes the IP address indicated by the field. If the IP address is not included in the IP address-based identifier, log on to the Log Service console and modify the IP address. Wait for 1 minute and then view the value of this field.
logtail_version	The version of Logtail.
os	The version of the operating system.
update_time	The last startup time of Logtail.

Logtail operational log file (ilogtail.LOG)

The *ilogtail.LOG* file records the operational logs of Logtail. The severity levels of logs in ascending order include INFO, WARN, ERROR. Logs with the INFO severity level can be ignored.

If an exception occurs during log collection, troubleshoot the exception based on the exception type and Logtail operational logs. For more information, see How do I view Logtail collection errors?.

Note If you submit a ticket for a log collection error, upload Logtail operational logs.

- File path
  - In a Linux-based server: The file is stored in the */usr/local/ilogtail/ilogtail.LOG* directory.
  - Containers: The file is stored in the */usr/local/ilogtail/ilogtail.LOG* directory.
  - Windows
    - 64-bit: The file is stored in the *C:\Program Files (x86)\Alibaba\Logtail\ilogtail.LOG* directory.
    - 32-bit: The file is stored in the *C:\Program Files\Alibaba\Logtail\ilogtail.LOG* directory.
- Sample file

```
$tail /usr/local/ilogtail/ilogtail.LOG
[2018-09-13 01:13:59.024679]    [INFO]    [3155]    [build/release64/sls/ilogtail/elogtai
l.cpp:123]    change working dir:/usr/local/ilogtail/
[2018-09-13 01:13:59.025443]    [INFO]    [3155]    [build/release64/sls/ilogtail/AppConf
ig.cpp:175]    load logtail config file, path:/etc/ilogtail/conf/ap-southeast-2/ilogtail_
config.json
[2018-09-13 01:13:59.025460]    [INFO]    [3155]    [build/release64/sls/ilogtail/AppConf
ig.cpp:176]    load logtail config file, detail:{
   "config_server_address" : "http://logtail.ap-southeast-2-intranet.log.aliyuncs.com",
   "data_server_list" : [
      {
         "cluster" : "ap-southeast-2",
         "endpoint" : "ap-southeast-2-intranet.log.aliyuncs.com"
      }
]
```

Operational logs of Logtail plug-ins (logtail_plugin.LOG)

The *logtail_plugin.LOG* file records the operational logs of Logtail plug-ins. The severity levels of logs in ascending order include INFO, WARN, ERROR. Logs with the INFO severity level can be ignored.

If an exception such as **CANAL_RUNTIME_ALARM** occurs, you can troubleshoot the exception based on the *logtail_plugin.LOG* file.

> **Note** If you submit a ticket for a Logtail plug-in exception, upload Logtail operational logs.

- File path
  - In a Linux-based server: The file is stored in the */usr/local/ilogtail/logtail_plugin.LOG* directory.
  - Containers: The file is stored in the */usr/local/ilogtail/logtail_plugin.LOG* directory.
  - In a Windows-based server:
    - 64-bit: The file is stored in the *C:\Program Files (x86)\Alibaba\Logtail\logtail_plugin.LOG* directory.
    - 32-bit: The file is stored in the *C:\Program Files\Alibaba\Logtail\logtail_plugin.LOG* directory.
- Sample file

```
$tail /usr/local/ilogtail/logtail_plugin.LOG
2018-09-13 02:55:30 [INF] [docker_center.go:525] [func1] docker fetch all:start
2018-09-13 02:55:30 [INF] [docker_center.go:529] [func1] docker fetch all:stop
2018-09-13 03:00:30 [INF] [docker_center.go:525] [func1] docker fetch all:start
2018-09-13 03:00:30 [INF] [docker_center.go:529] [func1] docker fetch all:stop
2018-09-13 03:03:26 [INF] [log_file_reader.go:221] [ReadOpen] [##1.0##sls-zc-test-hz-pub$
docker-stdout-config,k8s-stdout]    open file for read, file:/logtail_host/var/lib/docker
/containers/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5b2d624/7f46afec6a14
de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5b2d624-json.log    offset:40379573    statu
s:794354-64769-40379963
2018-09-13 03:03:26 [INF] [log_file_reader.go:221] [ReadOpen] [##1.0##k8s-log-c12ba2028cf
b444238cd9ac1286939f0b$docker-stdout-config,k8s-stdout]    open file for read, file:/logt
ail_host/var/lib/docker/containers/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd34
10f5b2d624/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5b2d624-json.log    o
ffset:40379573    status:794354-64769-40379963
2018-09-13 03:04:26 [INF] [log_file_reader.go:308] [CloseFile] [##1.0##sls-zc-test-hz-pub
$docker-stdout-config,k8s-stdout]    close file, reason:no read timeout    file:/logtail_
host/var/lib/docker/containers/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5
b2d624/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5b2d624-json.log    offse
t:40379963    status:794354-64769-40379963
2018-09-13 03:04:27 [INF] [log_file_reader.go:308] [CloseFile] [##1.0##k8s-log-c12ba2028c
fb444238cd9ac1286939f0b$docker-stdout-config,k8s-stdout]    close file, reason:no read ti
meout    file:/logtail_host/var/lib/docker/containers/7f46afec6a14de39b59ee9cdfbfa8a70c2f
a26f1148b2e2f31bd3410f5b2d624/7f46afec6a14de39b59ee9cdfbfa8a70c2fa26f1148b2e2f31bd3410f5b
2d624-json.log    offset:40379963    status:794354-64769-40379963
2018-09-13 03:05:30 [INF] [docker_center.go:525] [func1] docker fetch all:start
2018-09-13 03:05:30 [INF] [docker_center.go:529] [func1] docker fetch all:stop
```

Container path mapping file (docker_path_config.json)

The *docker_path_config.json* file is created when you collect container logs. The file records the path mapping between container log files and host log files. The file type is JSON.

If the **DOCKER_FILE_MAPPING_ALARM** message appears when you troubleshoot a log collection exception, it indicates that Docker files fail to be mapped to host files. You can use the *docker_path_config.json* file to troubleshoot the exception.

> **Note**
>
> - The *docker_path_config.json* file cannot be modified. If you delete the file, another file is automatically created. This action does not affect your business.
> - If you submit a ticket for a container log collection exception, upload this file.

- File path

  /usr/local/ilogtail/docker_path_config.json

- Sample file

```
$cat /usr/local/ilogtail/docker_path_config.json
{
   "detail" : [
      {
         "config_name" : "##1.0##k8s-log-c12ba2028cfb444238cd9ac1286939f0b$nginx",
         "container_id" : "df19c06e854a0725ea7fca7e0378b0450f7bd3122f94fe3e754d8483fd330d
10",
         "params" : "{\n   \"ID\" : \"df19c06e854a0725ea7fca7e0378b0450f7bd3122f94fe3e754
d8483fd330d10\",\n   \"Path\" : \"/logtail_host/var/lib/docker/overlay2/947db346695a1f65e
63e582ecfd10ae1f57019a1b99260b6c83d00fcd1892874/diff/var/log\",\n   \"Tags\" : [\n      \
"nginx-type\",\n      \"access-log\",\n      \"_image_name_\",\n      \"registry.cn-hangz
hou.aliyuncs.com/log-service/docker-log-test:latest\",\n      \"_container_name_\",\n
\"nginx-log-demo\",\n      \"_pod_name_\",\n      \"nginx-log-demo-h2lzc\",\n      \"_nam
espace_\",\n      \"default\",\n      \"_pod_uid_\",\n      \"87e56ac3-b65b-11e8-b172-001
63f008685\",\n      \"_container_ip_\",\n      \"172.20.4.224\",\n      \"purpose\",\n
\"test\"\n   ]\n}\n"
      }
   ],
   "version" : "0.1.0"
}
```

3.2. Select a network type

This topic describes how to select a network type when you use Logtail to collect logs.

Network type

- Alibaba Cloud internal network: The Alibaba Cloud internal network provides gigabits of shared bandwidth. Transmission of log data over the internal network is stable and fast. The Alibaba Cloud internal network includes virtual private clouds (VPCs) and the classic network.
- Internet: Transmission of log data over the Internet is limited by the network bandwidth. In addition, network issues such as jitters, latency, and packet loss may affect the speed and stability of data transmission.
- Global acceleration: This network type can be used to accelerate log collection by using the nodes of Alibaba Cloud Content Delivery Network (CDN). Compared with the Internet, global acceleration provides a more stable network with lower transmission latency.

Select a network type

- Internal network

  You can use the Alibaba Cloud internal network to transmit log data from an Elastic Compute Service (ECS) instance to a Log Service project. Examples are provided in the following two scenarios:

  - The ECS instance and the Log Service project belong to the same Alibaba Cloud account and reside in the same region.
  - The ECS instance and the Log Service project belong to different Alibaba Cloud accounts but reside in the same region.

  We recommend that you create a Log Service project in the region where the ECS instance resides. Then, you can collect logs of the ECS instance and send the logs to Log Service over the Alibaba Cloud internal network.

- Internet

  You can use the Internet to transmit log data from an ECS instance or server to a Log Service project. Examples are provided in the following two scenarios:

  - The ECS instance and the Log Service project reside in different regions.
  - The server is provided by a third-party cloud service provider or deployed in your self-managed data center.

- Global acceleration

  If your servers are deployed in your self-managed data center outside China or provided by a third-party cloud service provider outside China, you can enable global acceleration. Global acceleration can resolve stability and latency issues during data transmission over the Internet. For more information, see Global acceleration.

Server type	Reside in the same region as the project	Require an Alibaba Cloud account ID ①	Network type
ECS instance of the current Alibaba Cloud account	Yes	No	Alibaba Cloud internal network
	No	No	Internet or global acceleration
ECS instance of another Alibaba Cloud account	Yes	Yes	Alibaba Cloud internal network
	No	Yes	Internet or global acceleration
A server that is provided by a third-party cloud service provider or deployed in your self-managed data center	N/A	Yes	Internet or global acceleration

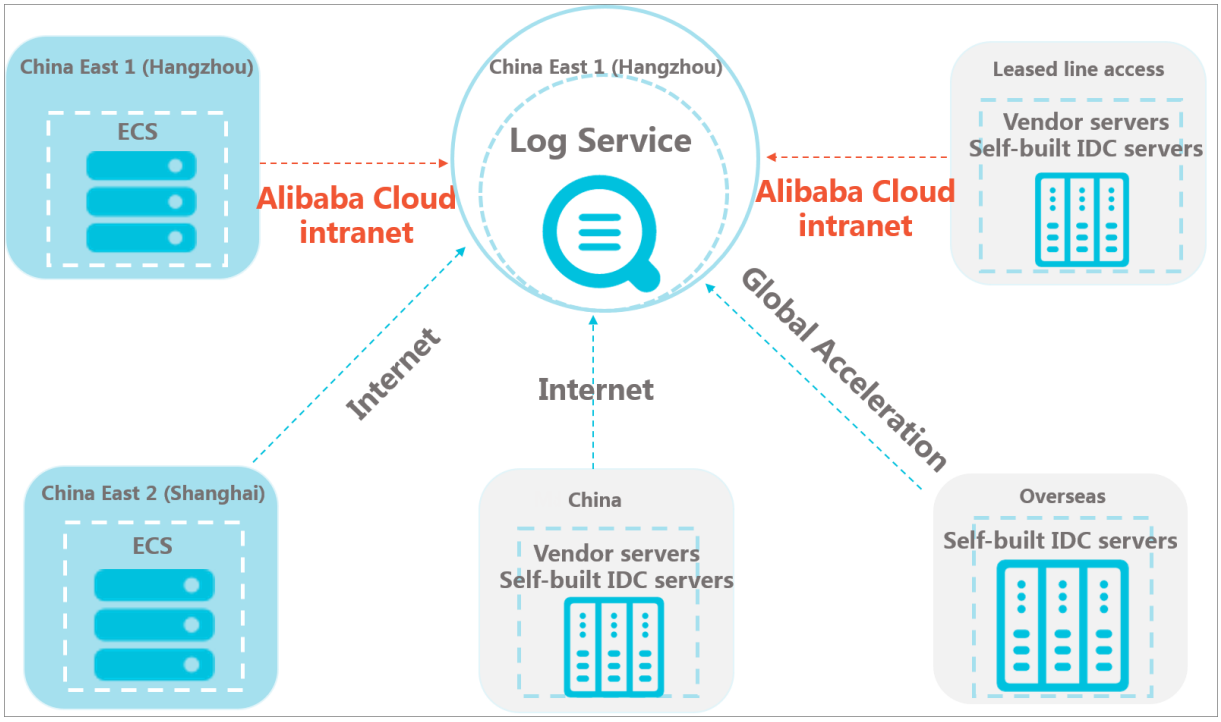
① If your server is an ECS instance that belongs to another Alibaba Cloud account, a server that is provided by a third-party cloud service provider, or a self-managed data center, you must first install Logtail on the server before you can collect logs from the server. Then, you must specify the ID of the Alibaba Cloud account for which Log Service is activated as a user identifier on the server. This way, the Alibaba Cloud account can use Logtail to collect logs from the server. For more information, see Configure a user identifier.

Example

The following table describes how to select a network type based on your business scenario.

> **Note** In this example, your Log Service project resides in the China (Hong Kong) region and your server resides in a global self-managed data center inside or outside China. We recommend that you select global acceleration in the China (Hong Kong) region to collect logs when you install Logtail on the server. Compared with the Internet, global acceleration provides a more stable network with low transmission latency.

Scenario	Region of the Log Service project	Server type	Region of the ECS instance	Region selected when Logtail is installed	Network type	Require an Alibaba Cloud account ID
The ECS instance and the Log Service project are in the same region.	China (Hangzhou)	ECS instance of the current Alibaba Cloud account	China (Hangzhou)	China (Hangzhou)	Internal network	No
The ECS instance and the Log Service project reside in different regions.	China (Shanghai)	ECS instance of the current Alibaba Cloud account	China (Beijing)	China (Shanghai)	Internet	No
The ECS instance and the Log Service project belong to different Alibaba Cloud accounts.	China (Shanghai)	ECS instance of another Alibaba Cloud account	China (Beijing)	China (Shanghai)	Internet	Yes
The server is deployed in your self-managed data center.	China (Shenzhen)	Server in the self-managed data center	N/A	China (Shenzhen)	Internet	Yes
Global acceleration is used.	China (Hong Kong)	Server in the self-managed data center	N/A	China (Hong Kong)	Global acceleration	Yes



3.3. Install

3.3.1. Install Logtail on ECS instances

Log Service allows you to install Logtail on Alibaba Cloud Elastic Compute Service (ECS) instances. This topic describes how to specify ECS instances and install Logtail on the ECS instances in the wizard that is used to create a Logtail configuration.

Prerequisites

If you use a RAM user to install Logtail on ECS instances, the RAM user must have the following permissions:

- AliyunOOSFullAccess permission: For more information about how to grant the AliyunOOSFullAccess permission to a RAM user, see Grant permissions to a RAM user.
- Custom permissions: To grant the following custom permissions to a RAM user, you must create a custom policy and attach the custom policy to the RAM user. For more information, see Create a custom policy and Grant permissions to a RAM user.

```
{
    "Version": "1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecs:DescribeTagKeys",
                "ecs:DescribeTags",
                "ecs:DescribeInstances",
                "ecs:DescribeInvocationResults",
                "ecs:RunCommand",
                "ecs:DescribeInvocations"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "oos:ListTemplates",
                "oos:StartExecution",
                "oos:ListExecutions",
                "oos:GetExecutionTemplate",
                "oos:ListExecutionLogs",
                "oos:ListTaskExecutions"
            ],
            "Resource": "*"
        }
    ]
}
```

Context

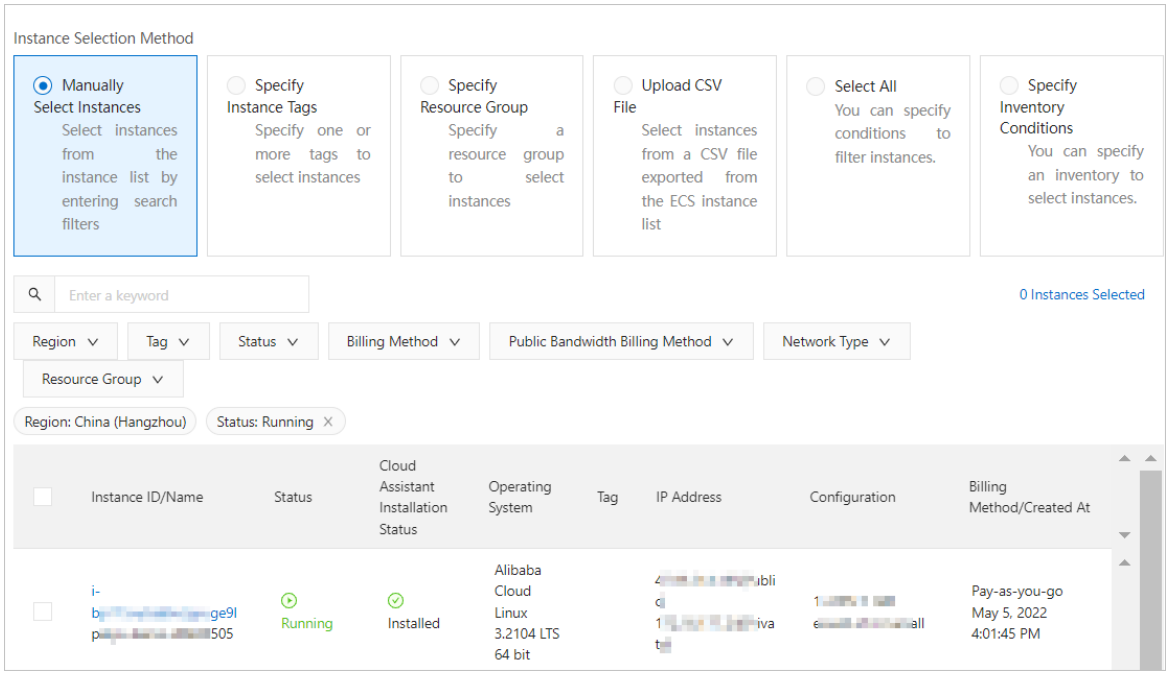
Log Service allows you to use Logtail to collect logs from ECS instances, self-managed servers, and servers that are provided by third-party cloud service providers. Before you collect logs, you must install Logtail on your servers. For more information about how to install Logtail on self-managed servers or servers that are provided by third-party cloud service providers, see Install Logtail on a Linux server or Install Logtail on a Windows server.

Procedure

1.
2. In the **Import Data** section, select **Delimiter Mode - Text Log**.

   In this example, Delimiter Mode - Text Log is selected as the data source. You can select a different data source based on your business requirements.
3. Select a project and a Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, click the **ECS Instances** tab.
5. Select ECS instances.

   You can use one of the following methods to select ECS instances.



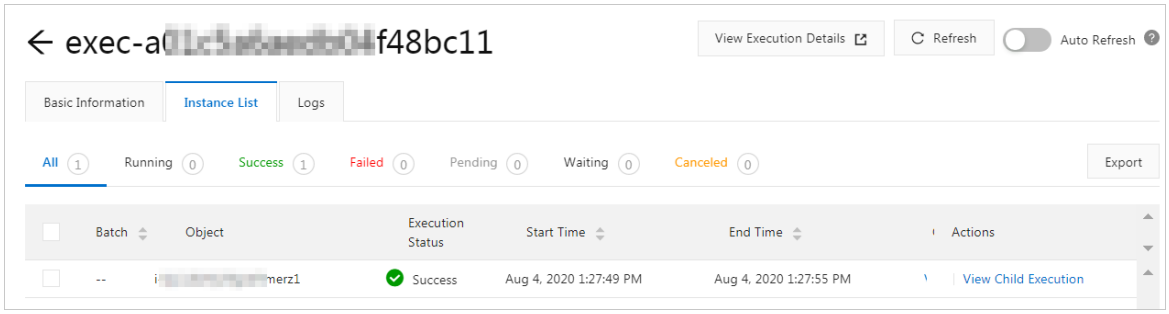
Method	Description
Manually Select Instances	From the instance list, you can select one or more instances on which you want to install Logtail. You can filter instances by Region, Tag, Status, Billing Method, and Network Type. You can also filter instances by using keywords.
Specify Instance Tags	You can add tags to ECS instances. This helps you categorize and manage resources. For more information, see Create or bind a tag. After you add tags, you can specify one or more tags to select instances. The system installs Logtail on all ECS instances to which the tags are added. A tag is a key-value pair. The tag key is required, and the tag value is optional. If you select only tag keys, the system installs Logtail on all ECS instances for which tag keys are specified.
Specify Resource Group	You can configure resource groups for ECS instances to manage resources in a hierarchical manner. For more information, see Resource groups. After you configure resource groups, you can select a resource group. The system installs Logtail on all ECS instances in the resource group.
Upload CSV File	In the ECS console, you can export a resource list to a CSV file. After you export the resource list, you can upload the CSV file for this parameter. The system installs Logtail on all ECS instances in the resource list.
Select All	If you select this option, the system automatically installs Logtail on all ECS instances that belong to the current Alibaba Cloud account.
Specify Inventory Conditions	After you create a configuration list in the Operation Orchestration Service (OOS) console, you can use the configuration list to specify the ECS instances on which you want to install Logtail. For more information, see Use OOS to collect the data of ECS instances.

6. (Optional)Configure advanced parameters.

Parameter	Description
Description	Enter a description for the installation task.
Tags for Resource Groups	After you select a resource group, the system adds the resource group information to the execution record.
Execution Mode	Select one of the following modes: ○ Automatic: If Logtail fails to be installed on a server, Log Service attempts to install Logtail on the next available server. ○ Suspend upon Failure: If Logtail fails to be installed on a server, Log Service attempts to install Logtail on the server again.
Rate Control	Select one of the following modes: ○ Concurrency-based Control: Logtail is installed on all of the specified ECS instances at the same time. ○ Batch-based Control: Logtail is installed on the specified ECS instances in batches.
Concurrency	Specify the concurrency rate. The value can be a number or percentage. For example, if you set the Concurrency parameter to 2 Targets, Logtail is installed on two ECS instances at the same time. **Note** If you set the **Rate Control** parameter to **Concurrency-based Control**, you must configure this parameter.
Batch Array	Enter an array that specifies how Logtail is installed. Log Service installs Logtail on the specified ECS instances in batches based on the array that you specify. Log Service installs Logtail on the next batch of ECS instances only after Logtail is installed on the current batch of ECS instances. You can enter an array that consists of numbers or percentages. If you set the Batch Array parameter to [1, 5%, 10%], Logtail is installed on an ECS instance in the first batch. In the second batch, Logtail is installed on 5% of the total instances. In the third and subsequent batches, Logtail is installed on 10% of the total instances. **Note** If you set the **Rate Control** parameter to **Batch-based Control**, you must configure this parameter.
Error Threshold	Specify the maximum number of errors or the maximum error rate that is allowed before the installation task is stopped. Default value: 0. For example, Logtail needs to be installed on four ECS instances. In this example, the Concurrency parameter is set to 1 Targets, and the Error Threshold parameter is set to 0 Errors. If Logtail fails to be installed on one of the ECS instances, the installation on the subsequent servers is canceled.

7. Click **Execute Now**.

   On the page that appears, you can view the installation results. The installation results include the basic information, instance list, and logs.



What's next

Configure the parameters in the Machine Group Settings and Logtail Config steps. For more information, see Collect text logs.

3.3.2. Install Logtail on a Linux server

This topic describes how to install, update, and uninstall Logtail on a Linux server.

Prerequisites

- At least one Linux server is available.
- The type of the network that is required to collect logs is determined. You can determine the network type based on the type of the server on which you want to install Logtail and the region where the server resides. For more information, see Select a network type.

Supported operating systems

- You can install Logtail on servers that run one of the following x86-64 Linux operating systems:
  - Alibaba Cloud Linux 2
  - Red Hat Enterprise Linux 6, Red Hat Enterprise Linux 7, and Red Hat Enterprise Linux 8
  - CentOS Linux 6, CentOS Linux 7, and CentOS Linux 8
  - Debian GNU/Linux 8, Debian GNU/Linux 9, Debian GNU/Linux 10, and Debian GNU/Linux 11
  - Ubuntu 14.04, Ubuntu 16.04, Ubuntu 18.04, and Ubuntu 20.04
  - SUSE Linux Enterprise Server 11, SUSE Linux Enterprise Server 12, and SUSE Linux Enterprise Server 15
  - openSUSE Leap 15.1, openSUSE Leap 15.2, and openSUSE Leap 42.3
  - Linux operating systems based on GNU C Library version 2.5 or later
- You can install Logtail on servers that run one of the following ARM64 Linux operating systems:
  - Alibaba Cloud Linux 3.2 for ARM
  - Anolis OS 8.2 for ARM or later
  - CentOS Linux 8.4 for ARM
  - Ubuntu 20.04 for ARM
  - Debian GNU/Linux 11.2 for ARM

Usage notes

- If your Logtail version is V0.0, such as Logtail V0.16.x, you can run the installation command in this topic to install Logtail on a Linux server. If you want to install Logtail V1.0 or update the Logtail version to Logtail V1.0, you must add the version number to the installation or update command. For example, if you want to install Logtail V1.0, you must run the `./logtail.sh install cn-hangzhou -v v1` command. If you want to update the Logtail version to Logtail V1.0, you must run the `sudo ./logtail.sh upgrade -v v1` command.
- If you run the installation command on a server on which Logtail is installed, the installer uninstalls Logtail from the server, deletes the */usr/local/ilogtail* directory, and then reinstalls Logtail. If the installation is successful, Logtail automatically runs and is added as a startup program.
- If the installation fails, submit a ticket.
- If you install Logtail on an Elastic Compute Service (ECS) instance that resides in the classic network and then change the network type from classic network to Virtual Private Cloud (VPC), you must update the Logtail configuration. For more information, see How do I update a Logtail configuration after I switch the network type of an ECS instance from the classic network to a VPC?.

Installation methods

The installation command varies based on the network type of the server on which you want to install Logtail.

- Alibaba Cloud internal network (classic network or VPC)
- Internet
- Global acceleration
- Install Logtail offline

Before you run the installation command, you must replace the *${your_region_name}* variable in the command with the value for the region where your project resides. The following table describes the value of the *${your_region_name}* variable for each region.

Region names for Logtail installation

Region	Value of ${your_region_name}
China (Hangzhou)	cn-hangzhou
China (Shanghai)	cn-shanghai
China (Qingdao)	cn-qingdao
China (Beijing)	cn-beijing
China (Zhangjiakou)	cn-zhangjiakou
China (Hohhot)	cn-huhehaote
China (Ulanqab)	cn-wulanchabu
China (Shenzhen)	cn-shenzhen
China (Heyuan)	cn-heyuan
China (Guangzhou)	cn-guangzhou

Region	Value of ${your_region_name}
China (Chengdu)	cn-chengdu
China (Hong Kong)	cn-hongkong
Russia (Moscow)	rus-west-1
US (Silicon Valley)	us-west-1
US (Virginia)	us-east-1
Singapore (Singapore)	ap-southeast-1
Australia (Sydney)	ap-southeast-2
Malaysia (Kuala Lumpur)	ap-southeast-3
Indonesia (Jakarta)	ap-southeast-5
Philippines (Manila)	ap-southeast-6
Thailand (Bangkok)	ap-southeast-7
India (Mumbai)	ap-south-1
Japan (Tokyo)	ap-northeast-1
South Korea (Seoul)	ap-northeast-2
Germany (Frankfurt)	eu-central-1
UAE (Dubai)	me-east-1
UK (London)	eu-west-1

Alibaba Cloud internal network (classic network or VPC)

- If you cannot identify the region where your ECS instance resides, you can use the auto parameter in the Logtail installation script to install Logtail.

  After you configure the auto parameter in the installation command, the Logtail installation script obtains and uses the metadata of the ECS instance to identify the region where your ECS instance resides. For more information about the metadata of ECS instances, see Overview of ECS instance metadata.

  i. Download the Logtail installation script over the Internet.

  The download consumes approximately 10 KB of Internet traffic.

  ```
  wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/linux64/logtail.sh -O logtail.sh;chmod 755 logtail.sh
  ```

  ii. Install Logtail by using the auto parameter.

The Logtail installation package for the region is automatically downloaded. The download does not consume Internet traffic.

```
./logtail.sh install auto
```

- If you can identify the region where your ECS instance resides, you can specify the region in the installation command.

  The Logtail installation script is downloaded over an internal network, and Logtail is manually installed. The download does not consume Internet traffic.

  i. Obtain the value of the *${your_region_name}* variable for the region where your Log Service project resides.

  For more information about the value of the *${your_region_name}* variable for each region, see Region names for Logtail installation. For example, the value of the *${your_region_name}* variable for the China (Hangzhou) region is cn-hangzhou.

  ii. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the installation command.

```
wget http://logtail-release-${your_region_name}.oss-${your_region_name}-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ${your_region_name}
```

The following table describes the installation commands used to install Logtail in each region where a Log Service project resides.

Region	Installation command
China (Hangzhou)	`wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hangzhou`
China (Shanghai)	`wget http://logtail-release-cn-shanghai.oss-cn-shanghai-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shanghai`
China (Qingdao)	`wget http://logtail-release-cn-qingdao.oss-cn-qingdao-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-qingdao`
China (Beijing)	`wget http://logtail-release-cn-beijing.oss-cn-beijing-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-beijing`

Region	Installation command
China (Zhangjiakou)	`wget http://logtail-release-cn-zhangjiakou.oss-cn-zhangjiakou-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-zhangjiakou`
China (Hohhot)	`wget http://logtail-release-cn-huhehaote.oss-cn-huhehaote-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-huhehaote`
China (Ulanqab)	`wget http://logtail-release-cn-wulanchabu.oss-cn-wulanchabu-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-wulanchabu`
China (Shenzhen)	`wget http://logtail-release-cn-shenzhen.oss-cn-shenzhen-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shenzhen`
China (Heyuan)	`wget http://logtail-release-cn-heyuan.oss-cn-heyuan-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-heyuan`
China (Guangzhou)	`wget http://logtail-release-cn-guangzhou.oss-cn-guangzhou-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-guangzhou`
China (Chengdu)	`wget http://logtail-release-cn-chengdu.oss-cn-chengdu-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-chengdu`
China (Hong Kong)	`wget http://logtail-release-cn-hongkong.oss-cn-hongkong-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hongkong`
US (Silicon Valley)	`wget http://logtail-release-us-west-1.oss-us-west-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-west-1`

Region	Installation command
US (Virginia)	`wget http://logtail-release-us-east-1.oss-us-east-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-east-1`
Singapore (Singapore)	`wget http://logtail-release-ap-southeast-1.oss-ap-southeast-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-1`
Australia (Sydney)	`wget http://logtail-release-ap-southeast-2.oss-ap-southeast-2-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-2`
Malaysia (Kuala Lumpur)	`wget http://logtail-release-ap-southeast-3.oss-ap-southeast-3-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-3`
Indonesia (Jakarta)	`wget http://logtail-release-ap-southeast-5.oss-ap-southeast-5-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-5`
Philippines (Manila)	`wget http://logtail-release-ap-southeast-6.oss-ap-southeast-6-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-6`
Thailand (Bangkok)	`wget http://logtail-release-ap-southeast-7.oss-ap-southeast-7-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-7`
Japan (Tokyo)	`wget http://logtail-release-ap-northeast-1.oss-ap-northeast-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-1`

Region	Installation command
South Korea (Seoul)	`wget http://logtail-release-ap-northeast-2.oss-ap-northeast-2-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-2`
India (Mumbai)	`wget http://logtail-release-ap-south-1.oss-ap-south-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-south-1`
Germany (Frankfurt)	`wget http://logtail-release-eu-central-1.oss-eu-central-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-central-1`
UAE (Dubai)	`wget http://logtail-release-me-east-1.oss-me-east-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install me-east-1`
UK (London)	`wget http://logtail-release-eu-west-1.oss-eu-west-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-west-1`
Russia (Moscow)	`wget http://logtail-release-rus-west-1.oss-rus-west-1-internal.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install rus-west-1`

Internet

1. Obtain the value of the `${your_region_name}` variable for the region where your Log Service project resides.

   For more information about the value of the *${your_region_name}* variable for each region, see Region names for Logtail installation. For example, the value of the *${your_region_name}* variable for the China (Hangzhou) region is cn-hangzhou.

2. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the installation command.

   ```
   wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ${your_region_name}-internet
   ```

The following table describes the installation commands used to install Logtail in each region where a Log Service project resides.

Region	Installation command
China (Hangzhou)	`wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hangzhou-internet`
China (Shanghai)	`wget http://logtail-release-cn-shanghai.oss-cn-shanghai.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shanghai-internet`
China (Qingdao)	`wget http://logtail-release-cn-qingdao.oss-cn-qingdao.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-qingdao-internet`
China (Beijing)	`wget http://logtail-release-cn-beijing.oss-cn-beijing.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-beijing-internet`
China (Zhangjiakou)	`wget http://logtail-release-cn-zhangjiakou.oss-cn-zhangjiakou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-zhangjiakou-internet`
China (Hohhot)	`wget http://logtail-release-cn-huhehaote.oss-cn-huhehaote.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-huhehaote-internet`
China (Ulanqab)	`wget http://logtail-release-cn-wulanchabu.oss-cn-wulanchabu.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-wulanchabu-internet`
China (Shenzhen)	`wget http://logtail-release-cn-shenzhen.oss-cn-shenzhen.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shenzhen-internet`

Region	Installation command
China (Heyuan)	`wget http://logtail-release-cn-heyuan.oss-cn-heyuan.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-heyuan-internet`
China (Guangzhou)	`wget http://logtail-release-cn-guangzhou.oss-cn-guangzhou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-guangzhou-internet`
China (Chengdu)	`wget http://logtail-release-cn-chengdu.oss-cn-chengdu.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-chengdu-internet`
China (Hong Kong)	`wget http://logtail-release-cn-hongkong.oss-cn-hongkong.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hongkong-internet`
US (Silicon Valley)	`wget http://logtail-release-us-west-1.oss-us-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-west-1-internet`
US (Virginia)	`wget http://logtail-release-us-east-1.oss-us-east-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-east-1-internet`
Singapore (Singapore)	`wget http://logtail-release-ap-southeast-1.oss-ap-southeast-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-1-internet`
Australia (Sydney)	`wget http://logtail-release-ap-southeast-2.oss-ap-southeast-2.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-2-internet`

Region	Installation command
Malaysia (Kuala Lumpur)	`wget http://logtail-release-ap-southeast-3.oss-ap-southeast-3.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-3-internet`
Indonesia (Jakarta)	`wget http://logtail-release-ap-southeast-5.oss-ap-southeast-5.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-5-internet`
Philippines (Manila)	`wget http://logtail-release-ap-southeast-6.oss-ap-southeast-6.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-6-internet`
Thailand (Bangkok)	`wget http://logtail-release-ap-southeast-7.oss-ap-southeast-7.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-7-internet`
Japan (Tokyo)	`wget http://logtail-release-ap-northeast-1.oss-ap-northeast-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-1-internet`
South Korea (Seoul)	`wget http://logtail-release-ap-northeast-2.oss-ap-northeast-2.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-2-internet`
Germany (Frankfurt)	`wget http://logtail-release-eu-central-1.oss-eu-central-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-central-1-internet`
UAE (Dubai)	`wget http://logtail-release-me-east-1.oss-me-east-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install me-east-1-internet`

Region	Installation command
India (Mumbai)	`wget http://logtail-release-ap-south-1.oss-ap-south-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-south-1-internet`
UK (London)	`wget http://logtail-release-eu-west-1.oss-eu-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-west-1-internet`
Russia (Moscow)	`wget http://logtail-release-rus-west-1.oss-rus-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install rus-west-1-internet`

Global acceleration

1. Obtain the value of the ${your_region_name} variable for the region where your Log Service project resides.

   For more information about the value of the *${your_region_name}* variable for each region, see Region names for Logtail installation. For example, the value of the *${your_region_name}* variable for the China (Hangzhou) region is cn-hangzhou.

2. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the installation command.

   ```
   wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ${your_region_name}-acceleration
   ```

The following table describes the installation commands used to install Logtail in each region where a Log Service project resides.

Region	Installation command
China (Beijing)	`wget http://logtail-release-cn-beijing.oss-cn-beijing.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-beijing-acceleration`
China (Qingdao)	`wget http://logtail-release-cn-qingdao.oss-cn-qingdao.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-qingdao-acceleration`

Region	Installation command
China (Hangzhou)	`wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hangzhou-acceleration`
China (Shanghai)	`wget http://logtail-release-cn-shanghai.oss-cn-shanghai.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shanghai-acceleration`
China (Shenzhen)	`wget http://logtail-release-cn-shenzhen.oss-cn-shenzhen.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shenzhen-acceleration`
China (Heyuan)	`wget http://logtail-release-cn-heyuan.oss-cn-heyuan.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-heyuan-acceleration`
China (Guangzhou)	`wget http://logtail-release-cn-guangzhou.oss-cn-guangzhou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-guangzhou-acceleration`
China (Zhangjiakou)	`wget http://logtail-release-cn-zhangjiakou.oss-cn-zhangjiakou.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-zhangjiakou-acceleration`
China (Hohhot)	`wget http://logtail-release-cn-huhehaote.oss-cn-huhehaote.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-huhehaote-acceleration`
China (Ulanqab)	`wget http://logtail-release-cn-wulanchabu.oss-cn-wulanchabu.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-wulanchabu-acceleration`

Region	Installation command
China (Chengdu)	`wget http://logtail-release-cn-chengdu.oss-cn-chengdu.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-chengdu-acceleration`
China (Hong Kong)	`wget http://logtail-release-cn-hongkong.oss-cn-hongkong.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-hongkong-acceleration`
US (Silicon Valley)	`wget http://logtail-release-us-west-1.oss-us-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-west-1-acceleration`
US (Virginia)	`wget http://logtail-release-us-east-1.oss-us-east-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install us-east-1-acceleration`
Singapore (Singapore)	`wget http://logtail-release-ap-southeast-1.oss-ap-southeast-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-1-acceleration`
Australia (Sydney)	`wget http://logtail-release-ap-southeast-2.oss-ap-southeast-2.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-2-acceleration`
Malaysia (Kuala Lumpur)	`wget http://logtail-release-ap-southeast-3.oss-ap-southeast-3.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-3-acceleration`
Indonesia (Jakarta)	`wget http://logtail-release-ap-southeast-5.oss-ap-southeast-5.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-5-acceleration`

Region	Installation command
Philippines (Manila)	`wget http://logtail-release-ap-southeast-6.oss-ap-southeast-6.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-6-acceleration`
Thailand (Bangkok)	`wget http://logtail-release-ap-southeast-7.oss-ap-southeast-7.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-southeast-7-acceleration`
Japan (Tokyo)	`wget http://logtail-release-ap-northeast-1.oss-ap-northeast-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-1-acceleration`
South Korea (Seoul)	`wget http://logtail-release-ap-northeast-2.oss-ap-northeast-2.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-northeast-2-acceleration`
Germany (Frankfurt)	`wget http://logtail-release-eu-central-1.oss-eu-central-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-central-1-acceleration`
UAE (Dubai)	`wget http://logtail-release-me-east-1.oss-me-east-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install me-east-1-acceleration`
India (Mumbai)	`wget http://logtail-release-ap-south-1.oss-ap-south-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install ap-south-1-acceleration`
UK (London)	`wget http://logtail-release-eu-west-1.oss-eu-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install eu-west-1-acceleration`

Region	Installation command
Russia (Moscow)	`wget http://logtail-release-rus-west-1.oss-rus-west-1.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install rus-west-1-acceleration`

Install Logtail offline

1. Log on to a server that can be accessed over the Internet.
2. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the download command to download the installation script and installation package.

   For more information about the value of the *${your_region_name}* variable for each region, see Region names for Logtail installation. For example, the value of the *${your_region_name}* variable for the China (Hangzhou) region is cn-hangzhou.

   - Download the installation script

     ```
     wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/linux64/logtail.sh
     ```

   - Download the installation package (x86-64)

     ```
     wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/linux64/logtail-linux64.tar.gz
     ```

   - Download the installation package (ARM)

     ```
     wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/linux64/aarch64/logtail-linux64.tar.gz
     ```

3. Copy the installation script and installation package to the server on which you want to install Logtail.
4. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the installation command on the server on which you want to install Logtail.

   For more information about the value of the *${your_region_name}* variable for each region, see Region names for Logtail installation. Examples:

   - The value of the *${your_region_name}* variable for the China (Hangzhou) region whose resources can be accessed over an internal network is cn-hangzhou.
   - The value of the *${your_region_name}* variable for the China (Hangzhou) region whose resources can be accessed over the Internet is cn-hangzhou-internet.
   - The value of the *${your_region_name}* variable for the China (Hangzhou) region whose resources can be accessed by using global acceleration is cn-hangzhou-acceleration.

   ```
   chmod +x logtail.sh; ./logtail.sh install-local ${your_region_name}
   ```

> **Note** If you want to update Logtail offline, you can download the latest installation package and run the `chmod +x logtail.sh; ./logtail.sh upgrade-local` command.

View the version of Logtail

Go to the installation directory of Logtail and open the */usr/local/ilogtail/app_info.json* file. The value of the logtail_version field is the version of Logtail.

- Command

```
cat /usr/local/ilogtail/app_info.json
```

- Output

```
{
   "UUID" : "0DF18E97-0F2D-486F-B77F-*********",
   "hostname" : "david*******",
   "instance_id" : "F4FAFADA-F1D7-11E7-846C-00163E30349E_*********_1515129548",
   "ip" : "**********",
   "logtail_version" : "0.16.30",
   "os" : "Linux; 2.6.32-220.23.2.ali1113.el5.x86_64; #1 SMP Thu Jul 4 20:09:15 CST 2013;
x86_64",
   "update_time" : "2020-01-05 13:19:08"
}
```

Update Logtail online

You can use the Logtail installation script logtail.sh to update Logtail. The installation script automatically selects an update method based on the configurations of Logtail that is installed.

> **Note** Logtail is temporarily stopped during the update. However, no log data is lost. After Logtail is updated, only the configuration files and the checkpoint files are retained. Other files are overwritten.

1. Run the following command to update Logtail:

```
wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/linux64/logtail.sh
-O logtail.sh; chmod 755 logtail.sh
sudo ./logtail.sh upgrade
```

2. Check the update result.

   If information similar to the following example is returned, the update is successful:

```
Stop logtail successfully.
ilogtail is running
Upgrade logtail success
{
   "UUID" : "***",
   "hostname" : "***",
   "instance_id" : "***",
   "ip" : "***",
   "logtail_version" : "0.16.30",
   "os" : "Linux; 3.10.0-693.2.2.el7.x86_64; #1 SMP Tue Sep 12 22:26:13 UTC 2017; x86_6
4",
   "update_time" : "2020-08-29 15:01:36"
}
```

Update Logtail offline

1. Log on to a server that can be accessed over the Internet.
2. Replace the *${your_region_name}* variable in the command with the value for the region where your project resides, and then run the download command to download the installation script and installation package.

   For more information about the value of the *${your_region_name}* variable for each region whose resources can be accessed over the Internet, see Region names for Logtail installation. For example, the value of the *${your_region_name}* variable for the China (Hangzhou) region whose resources can be accessed over the Internet is cn-hangzhou-internet.

   ```
   wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/li
   nux64/logtail.sh
   ```

   ```
   wget http://logtail-release-${your_region_name}.oss-${your_region_name}.aliyuncs.com/li
   nux64/logtail-linux64.tar.gz
   ```
3. Copy the installation script and installation package to the server on which you want to update Logtail.
4. Run the following command to update Logtail:

   ```
   chmod +x logtail.sh; ./logtail.sh upgrade-local
   ```

Start and stop Logtail

- Start Logtail

  Run the following command as the root user:

  ```
  /etc/init.d/ilogtaild start
  ```

- Stop Logtail

  Run the following command as the root user:

  ```
  /etc/init.d/ilogtaild stop
  ```

Uninstall Logtail

Run the following command to uninstall Logtail:

```
wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/linux64/logtail.sh -O
logtail.sh;chmod 755 logtail.sh;./logtail.sh uninstall
```

3.3.3. Install Logtail on a Windows server

This topic describes how to install Logtail on a Windows server.

Prerequisites

- At least one Windows server is available.
- The type of the network that is required to collect logs is determined. You can determine the network type based on the type of the server on which you want to install Logtail and the region where the server resides. For more information, see Select a network type.

Supported operating systems

You can install Logtail on servers that run one of the following Windows operating systems.

> **Note** Logtail supports Windows Server 2008 and Windows 7 that run on x86 or x86_64 and other Windows operating systems that run on x86_64.

- Microsoft Windows Server 2008
- Microsoft Windows Server 2012
- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows 7
- Microsoft Windows 10
- Microsoft Windows Server Version 1909
- Microsoft Windows Server Version 2004

Install Logtail

1. Download the installation package.
   - If the server resides in a region in the Chinese mainland, click Logtail installation package.
   - If the server resides in a region outside the Chinese mainland, click Logtail installation package.
2. Decompress the `logtail_installer.zip` package to the current directory.
3. Select an installation command based on the network type and the region where your Log Service project resides, and run the installation command.

   Run Windows PowerShell or Command Prompt as an administrator. Go to the `logtail_installer` directory and run the installation command. The directory contains the files that are extracted from the installation package.

Region	Alibaba Cloud internal network (classic network or VPC)	Internet	Global acceleration
China (Qingdao)	`.\logtail_installer.exe install cn-qingdao`	`.\logtail_installer.exe install cn-qingdao-internet`	`.\logtail_installer.exe install cn-qingdao-acceleration`
China (Beijing)	`.\logtail_installer.exe install cn-beijing`	`.\logtail_installer.exe install cn-beijing-internet`	`.\logtail_installer.exe install cn-beijing-acceleration`
China (Zhangjiakou)	`.\logtail_installer.exe install cn-zhangjiakou`	`.\logtail_installer.exe install cn-zhangjiakou-internet`	`.\logtail_installer.exe install cn-zhangjiakou-acceleration`
China (Hohhot)	`.\logtail_installer.exe install cn-huhehaote`	`.\logtail_installer.exe install cn-huhehaote-internet`	`.\logtail_installer.exe install cn-huhehaote-acceleration`
China (Ulanqab)	`.\logtail_installer.exe install cn-wulanchabu`	`.\logtail_installer.exe install cn-wulanchabu-internet`	`.\logtail_installer.exe install cn-wulanchabu-acceleration`
China (Hangzhou)	`.\logtail_installer.exe install cn-hangzhou`	`.\logtail_installer.exe install cn-hangzhou-internet`	`.\logtail_installer.exe install cn-hangzhou-acceleration`
China (Shanghai)	`.\logtail_installer.exe install cn-shanghai`	`.\logtail_installer.exe install cn-shanghai-internet`	`.\logtail_installer.exe install cn-shanghai-acceleration`
China (Shenzhen)	`.\logtail_installer.exe install cn-shenzhen`	`.\logtail_installer.exe install cn-shenzhen-internet`	`.\logtail_installer.exe install cn-shenzhen-acceleration`
China (Heyuan)	`.\logtail_installer.exe install cn-heyuan`	`.\logtail_installer.exe install cn-heyuan-internet`	`.\logtail_installer.exe install cn-heyuan-acceleration`
China (Guangzhou)	`.\logtail_installer.exe install cn-guangzhou`	`.\logtail_installer.exe install cn-guangzhou-internet`	`.\logtail_installer.exe install cn-guangzhou-acceleration`

Region	Alibaba Cloud internal network (classic network or VPC)	Internet	Global acceleration
China (Chengdu)	`.\logtail_installer.exe install cn-chengdu`	`.\logtail_installer.exe install cn-chengdu-internet`	`.\logtail_installer.exe install cn-chengdu-acceleration`
China (Hong Kong)	`.\logtail_installer.exe install cn-hongkong`	`.\logtail_installer.exe install cn-hongkong-internet`	`.\logtail_installer.exe install cn-hongkong-acceleration`
US (Silicon Valley)	`.\logtail_installer.exe install us-west-1`	`.\logtail_installer.exe install us-west-1-internet`	`.\logtail_installer.exe install us-west-1-acceleration`
US (Virginia)	`.\logtail_installer.exe install us-east-1`	`.\logtail_installer.exe install us-east-1-internet`	`.\logtail_installer.exe install us-east-1-acceleration`
Singapore (Singapore)	`.\logtail_installer.exe install ap-southeast-1`	`.\logtail_installer.exe install ap-southeast-1-internet`	`.\logtail_installer.exe install ap-southeast-1-acceleration`
Australia (Sydney)	`.\logtail_installer.exe install ap-southeast-2`	`.\logtail_installer.exe install ap-southeast-2-internet`	`.\logtail_installer.exe install ap-southeast-2-acceleration`
Malaysia (Kuala Lumpur)	`.\logtail_installer.exe install ap-southeast-3`	`.\logtail_installer.exe install ap-southeast-3-internet`	`.\logtail_installer.exe install ap-southeast-3-acceleration`
Indonesia (Jakarta)	`.\logtail_installer.exe install ap-southeast-5`	`.\logtail_installer.exe install ap-southeast-5-internet`	`.\logtail_installer.exe install ap-southeast-5-acceleration`
Philippines (Manila)	`.\logtail_installer.exe install ap-southeast-6`	`.\logtail_installer.exe install ap-southeast-6-internet`	`.\logtail_installer.exe install ap-southeast-6-acceleration`
Thailand (Bangkok)	`.\logtail_installer.exe install ap-southeast-7`	`.\logtail_installer.exe install ap-southeast-7-internet`	`.\logtail_installer.exe install ap-southeast-7-acceleration`

Region	Alibaba Cloud internal network (classic network or VPC)	Internet	Global acceleration
India (Mumbai)	`.\logtail_installer.exe install ap-south-1`	`.\logtail_installer.exe install ap-south-1-internet`	`.\logtail_installer.exe install ap-south-1-acceleration`
Japan (Tokyo)	`.\logtail_installer.exe install ap-northeast-1`	`.\logtail_installer.exe install ap-northeast-1-internet`	`.\logtail_installer.exe install ap-northeast-1-acceleration`
South Korea (Seoul)	`.\logtail_installer.exe install ap-northeast-2`	`.\logtail_installer.exe install ap-northeast-2-internet`	`.\logtail_installer.exe install ap-northeast-2-acceleration`
Germany (Frankfurt)	`.\logtail_installer.exe install eu-central-1`	`.\logtail_installer.exe install eu-central-1-internet`	`.\logtail_installer.exe install eu-central-1-acceleration`
UAE (Dubai)	`.\logtail_installer.exe install me-east-1`	`.\logtail_installer.exe install me-east-1-internet`	`.\logtail_installer.exe install me-east-1-acceleration`
UK (London)	`.\logtail_installer.exe install eu-west-1`	`.\logtail_installer.exe install eu-west-1-internet`	`.\logtail_installer.exe install eu-west-1-acceleration`
Russia (Moscow)	`.\logtail_installer.exe install rus-west-1`	`.\logtail_installer.exe install rus-west-1-internet`	`.\logtail_installer.exe install rus-west-1-acceleration`

Note Log Service cannot obtain the owner information about Elastic Compute Service (ECS) instances that belong to a different Alibaba Cloud account. In addition, Log Service cannot obtain the owner information about servers in data centers or servers from third-party cloud service providers. You must manually specify the IDs of Alibaba Cloud accounts as user identifiers for the servers after Logtail is installed. For more information, see Configure a user identifier.

Installation path

After you run the installation command, Logtail is automatically installed. The installation path varies based on the operating system. You cannot change the installation path.

- 32-bit Windows: *C:\Program Files\Alibaba\Logtail*

- 64-bit Windows: *C:\Program Files (x86)\Alibaba\Logtail*

> **Note** You can run 32-bit and 64-bit applications in a 64-bit Windows operating system. The operating system stores 32-bit applications in a separate x86 directory to ensure compatibility.
>
> Logtail for Windows is a 32-bit application and is installed in the *Program Files (x86)* directory in a 64-bit Windows operating system. If Logtail for 64-bit Windows is available, Logtail is installed in the *Program Files* directory by default.

View the version of Logtail

Go to the installation directory and open the *app_info.json* file. The value of the `logtail_version` field in the file is the Logtail version.

In the following example, the Logtail version is V1.0.0.0.

```
{
    "logtail_version" : "1.0.0.0"
}
```

Update Logtail

To update Logtail, you must download and decompress the most recent installation package. Then, you can follow the procedure for installing Logtail to complete the update. For more information, see Install Logtail.

> **Note** During update, Logtail is automatically uninstalled and then reinstalled. The original files in the installation directory are deleted. We recommend that you back up the files before you update Logtail.

Start and stop Logtail

1. Choose **Start Menu > Control Panel > Administrative Tools > Services**.
2. In the **Services** dialog box, select the service that you want to manage.
   - For Logtail V0.x.x.x, select LogtailWorker.
   - For Logtail V1.0.0.0 or later, select LogtailDaemon.
3. Right-click the service and select **Start**, **Stop**, or **Restart**.

Uninstall Logtail

Run Windows PowerShell or Command Prompt as an administrator. Go to the `logtail_installer` directory and run the following command to uninstall Logtail. The directory contains the files that are extracted from the installation package.

```
.\logtail_installer.exe uninstall
```

After Logtail is uninstalled, the Logtail installation directory is deleted. However, some configuration files are retained in the *C:\LogtailData* directory. You can manually delete the files based on your business requirements. The following configuration files are retained:

- *checkpoint*: contains information about the checkpoints that are generated by Logtail plug-ins. This file is generated only if the Logtail plug-ins are used.
- *user_config.d*: contains local collection configurations.

  JSON files in the directories are considered collection configurations. For example, */usr/local/ilogtail/user_log_config.json* is considered a collection configuration.
- *logtail_check_point*: contains information about the checkpoints that are generated by Logtail.
- *users*: contains the user identifier files that are configured.

3.3.4. Configure the startup parameters of Logtail

Log Service limits the collection performance of Logtail to prevent Logtail from consuming excessive server resources. If Logtail consumes excessive server resources, other services on the server may be affected. If you want to improve the collection performance of Logtail, you can modify the startup parameters of Logtail.

Scenarios

You can modify the startup parameters of Logtail in the following scenarios:

- You need to collect logs from a large number of log files and the log files occupy a large amount of memory. For example, you need to collect logs from more than 100 files or the log monitoring directory contains more than 5,000 log files.
- Log data is transmitted at a high speed, which causes high CPU utilization. For example, Logtail collects log data at a speed that exceeds 2 MB/s in simple mode and at a speed that exceeds 1 MB/s in full regex mode.
- Logtail sends data to Log Service at a speed that exceeds 10 MB/s.

Recommended parameter values

If you want to collect logs from JSON files, you can use the following parameter values that are obtained from real-world practice. The collection performance of Logtail in full regex mode and in delimiter mode is similar to the collection performance of Logtail in JSON mode. The collection performance of Logtail in simple mode is five times higher than the collection performance of Logtail in JSON mode. Both the complexity of data and rules and the numbers of directories and files from which you want to collect logs affect CPU utilization and memory usage. We recommend that you configure the following parameters based on the values in the table and your business requirements.

- Host environment

Parameter	Default collection speed	Collection speed higher than 10 MB/s	Collection speed higher than 20 MB/s	Collection speed higher than 40 MB/s
cpu_usage_limit	0.4	1	2	4
mem_usage_limit	384	1024	2048	4096
max-bytes-per-sec	20971520	209715200	209715200	209715200

Parameter	Default collection speed	Collection speed higher than 10 MB/s	Collection speed higher than 20 MB/s	Collection speed higher than 40 MB/s
process_thread_count	1	2	4	8
send_request_concurrency	4	20	40	80

- Container or Kubernetes environment

Environment variable	Default collection speed	Collection speed higher than 10 MB/s	Collection speed higher than 20 MB/s	Collection speed higher than 40 MB/s
cpu_usage_limit	2	3	5	9
mem_usage_limit	512	1024	2048	4096
max_bytes_per_sec	209715200	209715200	209715200	209715200
process_thread_count	1	2	4	8
send_request_concurrency	20	20	40	80
resources.limits.cpu	500M	1000M	2000M	4000M
resources.limits.memory	1 Gi	2 Gi	3 Gi	5 Gi

  If you want to collect logs from a container or a Kubernetes cluster, you can modify the startup parameters of Logtail by modifying DaemonSet-related environment variables. ConfigMaps are referenced by some environment variables, and the path to the ConfigMaps is **configmap > kube-system > alibaba-log-configuration**. You can also modify resources.limits.cpu and resources.limits.memory in **daemonset > kube-system > logtail-ds** to prevent the excessive usage of container resources.

If you configure the Logtail startup parameters based on the values of the Collection speed higher than 40 MB/s column in the preceding tables, the collection performance of Logtail approaches the upper limit. In this case, the performance does not significantly improve even if more threads are created. The following table describes the upper limit of the collection performance that Logtail can deliver in different collection modes.

> **Note** The actual collection performance may vary based on the test environment and the production environment.

Collection mode	Upper limit
Simple mode	440 MB/s
Full regex mode	70 MB/s
Delimiter mode	75 MB/s
JSON mode	75 MB/s

Configure the startup parameters of Logtail

1. Open the */usr/local/ilogtail/ilogtail_config.json* file on the server on which Logtail is installed.

   If you want to collect logs from a host, you can perform this step to configure the startup parameters of Logtail.

   If you want to collect logs from a container or a Kubernetes cluster, you can modify the startup parameters of Logtail by modifying DaemonSet-related environment variables. ConfigMaps are referenced by some environment variables, and the path to the ConfigMaps is **configmap > kube-system > alibaba-log-configuration**.

2. Configure the startup parameters of Logtail based on your business requirements.

   The following example shows the startup parameters of Logtail:

```
{
    ...
    "cpu_usage_limit" : 0.4,
    "mem_usage_limit" : 384,
    "max_bytes_per_sec" : 20971520,
    "process_thread_count" : 1,
    "send_request_concurrency" : 4,
    "buffer_file_num" : 25,
    "buffer_file_size" : 20971520,
    "buffer_file_path" : "",
    ...
}
```

> **Note**
>
> - The following table describes the commonly used startup parameters of Logtail. You can retain the default values for other startup parameters.
> - You can add or modify startup parameters based on your business requirements.

   Startup parameters of Logtail

Parameter	Type	Description	Example

Parameter	Type	Description	Example
cpu_usage_limit	double	The CPU utilization threshold for Logtail. The calculation is based on a single core. ◦ Valid values: 0.1 to the number of CPU cores of the current server ◦ Default value: 2 **Warning** cpu_usage_limit specifies a soft limit. The actual CPU utilization of Logtail may exceed the limit. If the CPU utilization of Logtail remains higher than this limit for 5 minutes, the system triggers a circuit breaker. Then, Logtail automatically restarts. For example, you set the parameter to 0.4. If the CPU utilization of Logtail remains higher than 40% for 5 minutes based on a single core, Logtail automatically restarts. In most cases, a single core supports a collection speed of 100 MB/s in simple mode and 20 MB/s in full regex mode.	"cpu_usage_limit" : 0.4

Parameter	Type	Description	Example
mem_usage_limit	int	The memory usage threshold for Logtail. Unit: MB. ◦ Valid values: 128 to 2048 ◦ Default value: 384 **Warning** mem_usage_limit specifies a soft limit. The actual memory usage of Logtail may exceed the limit. If the memory usage of Logtail remains higher than this limit for 5 minutes, the system triggers a circuit breaker. Then, Logtail automatically restarts. The following list describes how the setting of mem_usage_limit affects the number of files that can be monitored: ◦ If you use the default value, each Logtail configuration on a server allows Logtail to monitor up to 19,200 files. Logtail on a server can monitor up to 192,000 files. ◦ If you set the value to 2048, each Logtail configuration on a server allows Logtail to monitor up to 100,000 files. Logtail on a server can monitor up to 1,000,000 files. The maximum number of files that can be monitored is calculated by using the following formulas: ◦ Maximum number of files that can be monitored by Logtail for a Logtail configuration = Value of mem_usage_limit/100 × 5,000 ◦ Maximum number of files that can be monitored by Logtail on a server = Value of mem_usage_limit/100 × 50,000	"mem_usage_limit" : 384
max_bytes_per_sec	int	The highest speed at which Logtail sends raw data. Unit: bytes per second. ◦ Valid values: 1024 to 52428800 ◦ Default value: 20971520 For example, if you set the parameter to 2097152, the highest speed at which Logtail sends data is 2 MB/s. **Notice** If you set the parameter to a value that is greater than 20971520, the speed at which Logtail sends data is not limited. The value 20971520 indicates that the speed is 20 MB/s.	"max_bytes_per_sec" : 2097152

Parameter	Type	Description	Example
process_thread_count	int	The number of threads that are used by Logtail to process data. ◦ Valid values: 1 to 64 ◦ Default value: 1 In most cases, a thread provides a write speed of 24 MB/s in simple mode and 12 MB/s in full regex mode. We recommend that you retain the default value for this parameter.	"process_thread_count" : 1
send_request_concurrency	int	The maximum number of concurrent requests that can be sent by Logtail to asynchronously send data. ◦ Valid values: 1 to 1000 ◦ Default value: 20 If Log Service provides a high transactions per second (TPS), you can set this parameter to a larger value. Each concurrent request supports a network throughput of 0.5 MB/s to 1 MB/s. The actual network throughput for a concurrent request varies based on the network latency. **Note** If the value of this parameter is large, concurrent requests may occupy an excessive number of network ports. In this case, you must adjust the TCP-related parameters.	"send_request_concurrency" : 4
buffer_file_num	int	The maximum number of files that can be cached. ◦ Valid values: 1 to 100 ◦ Default value: 25 If a network error occurs or the limits of data write are reached, Logtail caches parsed logs to the local files in the installation directory. Logtail parses raw logs in real time. After the issues are fixed, Logtail retries to send the cached logs.	"buffer_file_num" : 25
buffer_file_size	int	The maximum size of a cached file. Unit: bytes. ◦ Valid values: 1048576 to 104857600 ◦ Default value: 20971520 The maximum disk space that can be occupied by cached files is calculated by multiplying the value of the buffer_file_size parameter by the value of the buffer_file_num parameter.	"buffer_file_size" : 20971520

Parameter	Type	Description	Example
buffer_file_path	String	The directory in which cached files are stored. This parameter is empty by default, which indicates that cached files are stored in the installation directory of Logtail. The default directory is */usr/local/ilogtail*. If you specify a value for this parameter, you must move the cached files whose name matches *logtail_buffer_file_** from the installation directory of Logtail to the directory that you specify. This way, Logtail can read, send, and then delete the cached files.	"buffer_file_path" : ""
bind_interface	String	The name of the network interface controller (NIC) that is associated with the server on which Logtail is installed. This parameter is empty by default, which indicates that the server is automatically associated with an available NIC. If you specify a value for this parameter, such as eth1, Logtail uses the NIC to upload logs. This parameter is available only if Logtail runs on a Linux server.	"bind_interface" : ""
check_point_filename	String	The path to the checkpoint files of Logtail. Default value: */tmp/logtail_check_point*.	"check_point_filename" : /tmp/logtail_check_point
check_point_dump_interval	int	The interval at which Logtail updates checkpoint files. Default value: 900. Unit: seconds. If you retain the default value, Logtail updates checkpoint files at 15-minute intervals. This parameter is available only for Logtail V1.0.19 or later.	"check_point_dump_interval" : 900
user_config_file_path	String	The path to the file that stores Logtail configurations. The file is named *user_log_config.json* and stored in the directory of the BIN file that is created for the Logtail process.	"user_config_file_path" : user_log_config.json
docker_file_cache_path	String	The path to the file that records the path mappings between container files and host files. By default, the path is */usr/local/ilogtail/docker_path_config.json*. This parameter is available only for Logtail V0.16.54 or later.	"docker_file_cache_path": /usr/local/ilogtail/docker_path_config.json
discard_old_data	Boolean	Specifies whether to discard historical logs. Default value: true. This value indicates that logs that were generated more than 12 hours before the current time are discarded.	"discard_old_data" : true

Parameter	Type	Description	Example
ilogtail_discard_interval	int	The time threshold for discarding logs. If the difference between the time at which the logs were generated and the current time exceeds the threshold, the logs are discarded. Default value: 43200. Unit: seconds. The value 43200 indicates that the threshold is 12 hours.	"ilogtail_discard_interval": 43200
working_ip	String	The server IP address that is reported by Logtail to Log Service. This parameter is empty by default, which indicates that Log Service automatically obtains the IP address of the server on which Logtail is installed.	"working_ip" : ""
working_hostname	String	The server hostname that is reported by Logtail to Log Service. This parameter is empty by default, which indicates that Log Service automatically obtains the hostname of the server on which Logtail is installed.	"working_hostname" : ""
max_read_buffer_size	long	The maximum size of a log that Logtail can read. Unit: bytes. Default value: 524288. The default value 524288 indicates that the maximum size is 512 KB. Maximum value: 4194304. The value 4194304 indicates that the maximum size is 4 MB. If the size of a log exceeds 524,288 bytes, you can change the value of this parameter.	"max_read_buffer_size" : 524288
oas_connect_timeout	long	The timeout period of the connection that is established by Logtail to send a request to obtain the Logtail configuration or AccessKey pair. Default value: 5. Unit: seconds. If the connections cannot be established before timeout due to poor network conditions, you can change the value of this parameter.	"oas_connect_timeout" : 5
oas_request_timeout	long	The timeout period of the request that is sent by Logtail to obtain the Logtail configuration or AccessKey pair. Default value: 10. Unit: seconds. If the connections cannot be established before timeout due to poor network conditions, you can change the value of this parameter.	"" : 10
data_server_port	long	If you set the data_server_port parameter to 443, Logtail transfers data to Log Service over HTTPS. This parameter is available only for Logtail V1.0.10 or later.	"data_server_port": 443

Parameter	Type	Description	Example
enable_log_time_auto_adjust	Boolean	If you set the enable_log_time_auto_adjust parameter to true, the log time is adapted to the local time of the server. To ensure data security, Log Service checks the time information in requests, including the requests sent by Logtail. This information indicates the time at which a request is sent. Log Service rejects requests that are sent 15 minutes earlier or later than the time in Log Service. The time information in a request is considered as the local time of the server. In some test scenarios, the local time must be changed to a future point in time. If you change the local time of the server, Log Service rejects the requests from Logtail, and data cannot be written to Log Service. You can use this parameter to adapt the log time to the local time of the server. This parameter is available only for Logtail V1.0.19 or later. **Notice** ◦ If you set the enable_log_time_auto_adjust parameter to true, the offset between the time in Log Service and the local time of the server is added to the log time. The offset is updated only if a request is rejected by Log Service. Therefore, the time of a log that is queried by Log Service may be different from the time at which the log is written. ◦ Part of the logic for Logtail changes based on the incremental increase of the system time. We recommend that you restart Logtail after you change the local time of the server.	"enable_log_time_auto_adjust": true
accept_multi_config	Boolean	Specifies whether to allow Logtail to collect data from the same file by using multiple Logtail configurations. Default value: false. This value indicates that Logtail cannot collect data from the same file by using multiple Logtail configurations. By default, Logtail can use only one Logtail configuration to collect data from a file. If you want to allow Logtail to collect data from a file by using multiple Logtail configurations, you can set this parameter to true. Each Logtail configuration has an independent collection process. If multiple Logtail configurations are used to collect data from the same file, the CPU utilization and memory usage increase. This parameter is available only for Logtail V0.16.26 or later.	"accept_multi_config": true

Parameter	Type	Description	Example
enable_checkpoint_sync_write	Boolean	Specifies whether to enable the sync write feature. Default value: false. This value indicates that the sync write feature is disabled. The sync write feature is used together with the ExactlyOnce write feature. After you enable the ExactlyOnce write feature, Logtail records fine-grained checkpoints by file to the disk of the server on which Logtail is installed. By default, Logtail does not call the sync function to write checkpoints to the disk. However, if buffered data fails to be written to the disk when the server restarts, the checkpoints may be lost. In this case, you can set the enable_checkpoint_sync_write parameter to true to enable the sync write feature. For more information, see Additional information: ExactlyOnce write feature. This parameter is available only for Logtail V1.0.20 or later.	"enable_checkpoint_sync_write": false
enable_env_ref_in_config	Boolean	Specifies whether to enable the environment variable replacement feature in Logtail configurations. Default value: false. After this feature is enabled, you can use `${xxx}` as the placeholder for the environment variable `xxx` when you create a Logtail configuration in the Log Service console. For example, if you set Log Path to `/${xxx}/logs` and the environment variable to `xxx=root`, the path to the file from which Logtail collects logs is `/root/logs`. If `${` and `}` are used in your Logtail configuration, you can use `$${` and `$}` to escape the characters. This parameter is available only for Logtail V1.0.31 or later.	"enable_env_ref_in_config": false
docker_config_update_interval	int	The minimum interval at which the container path is updated. Default value for versions earlier than Logtail V1.0.32: 10. Default value for Logtail V1.0.32 or later: 3. Unit: seconds. This parameter is used together with the max_docker_config_update_times parameter. If one of the values for the two parameters is reached, the container path is no longer updated.	"docker_config_update_interval": 3

Parameter	Type	Description	Example
max_docker_config_update_times	int	The maximum number of times that the container path can be updated within 3 minutes. Default value for versions earlier than Logtail V1.0.32: 3. Default value for Logtail V1.0.32 or later: 10. By default, if the container path is updated more than three times within a 3-minute period, the container path cannot be updated again until 3 minutes later.	"max_docker_config_update_times": 10

3. Restart Logtail for the new settings to take effect.

```
/etc/init.d/ilogtaild stop && /etc/init.d/ilogtaild start
```

After you restart Logtail, you can run the `/etc/init.d/ilogtaild status` command to check the status of Logtail.

Appendix: Environment variables

The following table describes the mappings between environment variables and the startup parameters of Logtail. For information about the startup parameters of Logtail, see Startup parameters of Logtail.

Mappings between environment variables and the startup parameters of Logtail

Parameter	Environment variable	Priority	Supported version
cpu_usage_limit	cpu_usage_limit	If you use environment variables and the configuration file named /usr/local/ilogtail/ilogtail_config.json to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.32 or later
mem_usage_limit	mem_usage_limit	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.32 or later

Parameter	Environment variable	Priority	Supported version
max_bytes_per_sec	max_bytes_per_sec	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.32 or later
process_thread_count	process_thread_count	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.32 or later
send_request_concurrency	send_request_concurrency	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.32 or later
check_point_filename	check_point_filename or ALIYUN_LOGTAIL_CHECK_POINT_PATH	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the environment variables take effect.	Logtail V0.16.36 or later
docker_file_cache_path	docker_file_cache_path	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.54 or later

Parameter	Environment variable	Priority	Supported version
user_config_file_path	user_config_file_path	If you use environment variables and the configuration file of Logtail to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
discard_old_data	discard_old_data	If you use environment variables and the configuration file of Logtail to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
working_ip	working_ip or ALIYUN_LOGTAIL_WORKING_IP	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
working_hostname	working_hostname or ALIYUN_LOGTAIL_WORKING_HOSTNAME	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
max_read_buffer_size	max_read_buffer_size	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later

Parameter	Environment variable	Priority	Supported version
oas_connect_timeout	oas_connect_timeout	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
oas_request_timeout	oas_request_timeout	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
data_server_port	data_server_port	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
accept_multi_config	accept_multi_config	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V0.16.56 or later
enable_log_time_auto_adjust	enable_log_time_auto_adjust	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V1.0.19 or later
check_point_dump_interval	check_point_dump_interval	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V1.0.19 or later

Parameter	Environment variable	Priority	Supported version
enable_checkpoint_sync_write	enable_checkpoint_sync_write	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V1.0.20 or later
docker_config_update_interval	docker_config_update_interval or ALIYUN_LOGTAIL_DOCKER_CONFIG_UPDATE_INTERVAL	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V1.0.29 or later
max_docker_config_update_times	max_docker_config_update_times or ALIYUN_LOGTAIL_MAX_DOCKER_CONFIG_UPDATE_TIMES	If you use environment variables and the configuration file to modify the startup parameters of Logtail, the modifications from the configuration file take effect.	Logtail V1.0.29 or later

3.4. Machine Group

3.4.1. Introduction

A machine group is a virtual group that contains multiple servers. Log Service uses machine groups to manage servers whose logs are to be collected by using Logtail.

You can create a machine group and add servers to the machine group in the Log Service console. Then you can create Logtail configurations for log collection and apply the configurations to the machine group. This way, you can collect logs from the servers based on the configurations.

To identify a machine group, you can use one of the following methods:

- IP address: uses the IP addresses of the servers in the machine group as the identifier of the machine group. Each server in the group can be identified by using its unique IP address.
- Custom ID: uses a custom ID to identify the machine group. Servers in the machine group have the same custom ID.

> **Note** You may want to collect logs from the servers that are provided by another cloud service provider, that reside in your on-premises data center, or belong to another Alibaba Cloud account. In this case, you must configure user identities for the servers before you add the servers to a machine group. For more information, see Configure a user identifier.

IP address-based machine groups

You can add the IP addresses of multiple servers to the identifier of a machine group. Then the servers are added to the machine group.

- If you want to collect logs of Elastic Compute Service (ECS) instances, you can add the private IP addresses of the instances to the identifier of the machine group. However, you must make sure that the mapping between hostnames and IP addresses is not configured and the network types of the instances are not changed.
- In other cases, add the IP addresses of the servers that Logtail automatically obtains after it is installed on the servers. The IP address of a server that Logtail obtains is indicated by the *ip* field. The field is recorded in the app_info.json file of the server. Logtail can obtain a server IP address in different scenarios.
  - If the hostname-to-IP address mapping is configured for the server in the */etc/hosts* file, Logtail obtains the mapped IP address.
  - If the hostname-to-IP address mapping is not configured for the server in the */etc/hosts* file of the server, Logtail obtains the IP address of the first network interface card.

> **Note** Log data may not be transferred over the Alibaba Cloud internal network even if the IP addresses that you configure in the identifier of a machine group are internal IP addresses. If you select the **Alibaba Cloud internal network (classic network or VPC)** mode when you install Logtail on an ECS instance, logs are collected by using the Alibaba Cloud internal network.

Custom ID-based machine groups

You can use a custom ID to identify a machine group in the following scenarios:

- If your servers reside in multiple custom network environments such as virtual private clouds (VPCs), some IP addresses of the servers may be the same. In this case, Logtail cannot collect logs as expected. You can use a custom ID to prevent this issue.
- Automatic scaling of a machine group. In this case, you only need to configure the same custom ID for new servers. Log Service identifies these servers and adds them to the machine group.

In most cases, a system consists of multiple modules. You can scale out each module by adding multiple servers to the module. To collect logs from the modules, you can create a machine group for each module. To identify the machine group of each module, you can create a custom ID for each machine group. For example, a website consists of an HTTP request processing module, a caching module, a logic processing module, and a storage module. The custom IDs of the machine groups that you create for these modules can be http_module, cache_module, logic_module, and store_module.

3.4.2. Configure a user identifier

This topic describes how to specify the ID of an Alibaba Cloud account as a user identifier on a server.

Prerequisites

- A server is available.

  The server can be an Elastic Compute Service (ECS) instance that belongs to another Alibaba Cloud account, a server that is provided by a third-party cloud service provider, or a self-managed data center.
- Logtail is installed on the server. For more information, see Install Logtail on a Linux server and Install

Logtail on a Windows server.

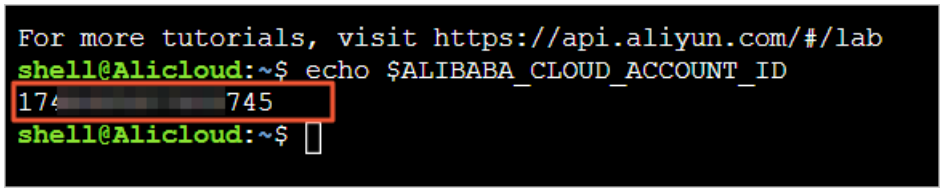
Context

If your server is an ECS instance that belongs to another Alibaba Cloud account, a server that is provided by a third-party cloud service provider, or a self-managed data center, you must first install Logtail on the server before you can collect logs from the server. Then, you must specify the ID of the Alibaba Cloud account for which Log Service is activated as a user identifier on the server. This way, the Alibaba Cloud account can use Logtail to collect logs from the server. If you do not configure a user identifier on your server, Log Service cannot receive the heartbeat of the server and Logtail cannot collect logs from the server.

Step 1: Obtain the ID of the Alibaba Cloud account for which Log Service is activated

1. Use the Alibaba Cloud account for which Log Service is activated to log on to the Alibaba Cloud official website.
2. Open Cloud Shell.
3. Run the following command to obtain the ID of the Alibaba Cloud account for which Log Service is activated:

```
echo $ALIBABA_CLOUD_ACCOUNT_ID
```



Step 2: Configure a user identifier

1. Log on to the server.
2. Specify the ID that you obtained as a user identifier on the server.

> **Notice**
> - If the */etc/ilogtail/users* directory does not exist, create the directory.
> - After you configure or delete a user identifier, the change takes effect within 1 minute.

- Linux

  In the */etc/ilogtail/users* directory, create a file and set the name of the file to the ID of the Alibaba Cloud account.

  ```
  touch /etc/ilogtail/users/17****745
  ```

- Windows

  In the *C:\LogtailData\users* directory, create a file and set the name of the file to the ID of the Alibaba Cloud account.

- Run the following command in Windows PowerShell to create the file:

```
ni C:\LogtailData\users\17******745
```

- Run the following command in the Command Prompt to create the file:

```
type nul > C:\LogtailData\users\17******745
```

Multiple Alibaba Cloud accounts

If you want to use Log Service that is activated for multiple Alibaba Cloud accounts to collect logs from the same server, you can create multiple user identifier files on the server and configure the IDs of the Alibaba Cloud accounts as user identifiers. Example:

```
touch /etc/ilogtail/users/17****742
touch /etc/ilogtail/users/17****743
```

Delete a user identifier

Notice We recommend that you delete user identifier files that you no longer need on a server at the earliest opportunity. Then, Log Service that is activated for the related Alibaba accounts has no permissions to collect logs from the server.

- Linux

  Run the following command to delete the specified user identifier file:

```
rm /etc/ilogtail/users/17****745
```

- Windows

  Run the following command to delete the specified user identifier file:

```
del C:\LogtailData\users\17******745
```

What's next

After you configure the ID of an Alibaba Cloud account as a user identifier on a server, you can create a machine group. For more information, see Create an IP address-based machine group or Create a custom ID-based machine group.

3.4.3. Create an IP address-based machine group

Log Service allows you to define a machine group by using IP addresses. This topic describes how to create an IP address-based machine group in the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and a Logstore.
- At least one server is available.
  - If you use an Elastic Compute Service (ECS) instance, you must make sure that the ECS instance belongs to the same Alibaba Cloud account and region as your Log Service.

- If you use an ECS instance that belongs to a different Alibaba Cloud account from your Log Service, you must configure a user identifier before you create a machine group. This also applies if you use a server in a data center or from a third-party cloud service provider. For more information, see Configure a user identifier.

- Logtail is installed on the servers. For more information, see Install Logtail on ECS instances.

Procedure

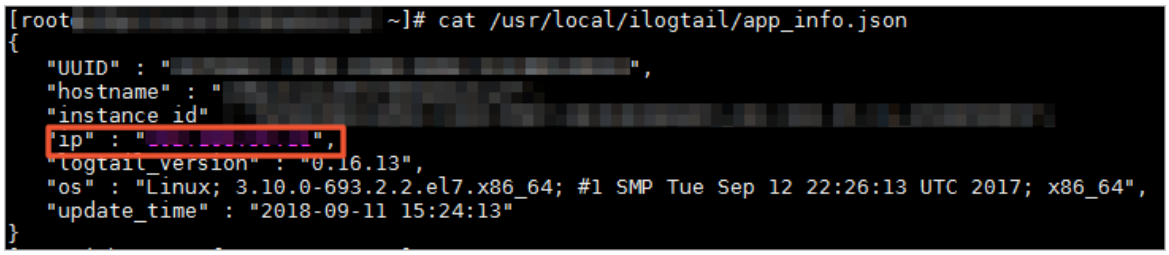
1. Obtain the IP addresses of servers.

   The IP addresses that are obtained by Logtail are recorded in the ip field of the *app_info.json* file.

   On the servers where Logtail is installed, you can go to the following paths to view the *app_info.json* file:

   - Linux: */usr/local/ilogtail/app_info.json*
   - 64-bit Windows: *C:\Program Files (x86)\Alibaba\Logtail\app_info.json*
   - 32-bit Windows: *C:\Program Files\Alibaba\Logtail\app_info.json*

   The following figure shows how to view the IP address of a Linux server.

2.
3. In the **Projects** section, click the project.
4. In the left navigation sidebar, choose Resources > **Machine Groups**.
5. On the right of **Machine Groups**, choose > **Create Machine Group**.
6. In the **Create Machine Group** panel, configure the parameters and click **OK**. The following table describes the parameters.

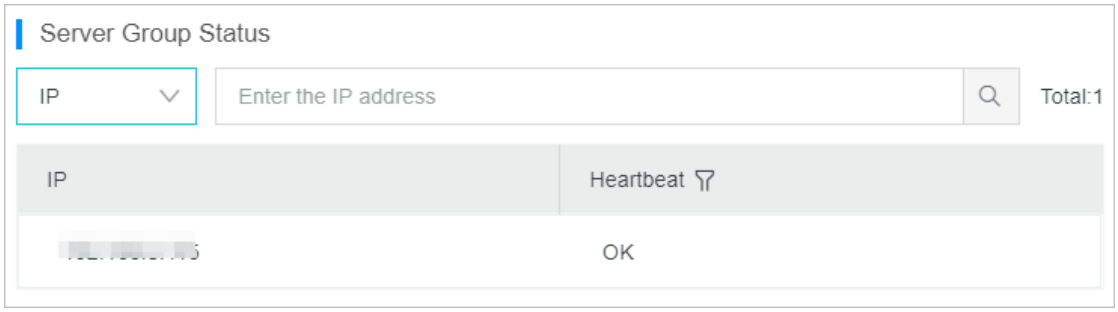
Parameter	Description
Name	Specify the name of the machine group. The name must be 3 to 128 characters in length, and can contain only lowercase letters, digits, hyphens (-), and underscores (_). The name must start and end with a lowercase letter or a digit. **Notice** After the machine group is created, you cannot change the name of the machine group. Proceed with caution.
Identifier	Select **IP Addresses**.

Parameter	Description
Topic	Specify the topic of the machine group. The topic is used to differentiate the logs that are generated by different servers. For more information, see Log topics.
IP Addresses	Enter the IP addresses of servers that you obtain in Step 1. **Note** ◦ If you want to add multiple servers to the machine group, separate the IP addresses with line feeds. ◦ Do not add Windows and Linux servers to the same machine group.

After you create the machine group, the machine group takes effect after approximately 2 minutes.

7. View the status of the machine group.
   i. In the Machine Groups list, click the machine group that you create.
   ii. On the **Machine Group Settings** page, view the server details and machine group status.

   If a value in the **Heartbeat** column is **OK**, the server is connected to Log Service. If the value is **FAIL**, the connection failed. For more information, see What do I do if no heartbeat connections are detected on Logtail?



3.4.4. Create a custom ID-based machine group

Log Service allows you to create a custom ID-based machine group. This topic describes how to create a custom ID-based machine group in the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and a Logstore.
- At least one server is available.
  - If you use an Elastic Compute Service (ECS) instance, you must make sure that the ECS instance belongs to the same Alibaba Cloud account and region as your Log Service.
  - If you use an ECS instance that belongs to a different Alibaba Cloud account from your Log Service, you must configure a user identifier before you create a machine group. This also applies if you use a server in a data center or from a third-party cloud service provider. For more information, see Configure a user identifier.

- Logtail is installed on the servers. For more information, see Install Logtail on ECS instances.

Context

Custom ID-based machine groups offer significant benefits in the following scenarios:

- If your servers reside in multiple custom network environments such as virtual private clouds (VPCs), some IP addresses of the servers may conflict. In this case, Logtail cannot collect logs as expected. You can use a custom ID to prevent this issue.
- If you want to add multiple servers to a machine group, you can set the same custom ID for new servers as the machine group. Log Service identifies the custom ID and adds the servers with the same custom ID to the machine group.

Procedure

1. Create a file named *user_defined_id* in a specified directory.
   - Linux: Store the file in the */etc/ilogtail/user_defined_id* directory.
   - Windows: Store the file in the *C:\LogtailData\user_defined_ided_id* directory.
2. Set a custom ID for the server.

   > **Note**
   > - Windows and Linux servers cannot be added to the same machine group. You cannot set the same custom ID for Linux and Windows servers.
   > - You can set one or more custom IDs for a single server and separate custom IDs with line feeds.
   > - In the Linux server, if the */etc/ilogtail/* directory or the */etc/ilogtail/user_defined_id* file does not exist, you can create the directory and file. In the Windows server, if the *C:\LogtailData* directory or the *C:\LogtailData\user_defined_id* file does not exist, you can also create the directory and file.

   - Linux:

     Set the custom ID in the */etc/ilogtail/user_defined_id* file. For example, if you want to set the custom ID to `userdefined`, run the following command to edit the file. In the file, enter `user defined`.

     ```
     vim /etc/ilogtail/user_defined_id
     ```

   - Windows:

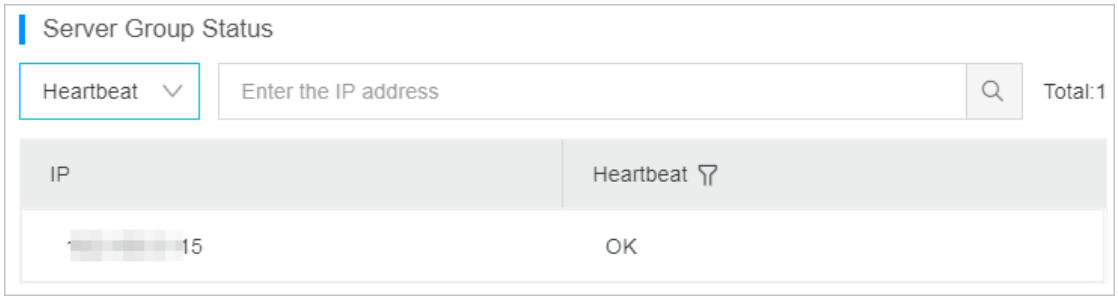
     Set the custom ID in the *C:\LogtailData\user_defined_id* file. For example, if you want to set the custom ID to `userdefined_windows`, enter `userdefined_windows` in the *C:\LogtailData\user_defined_id* file.
3.
4. In the **Projects** section, click the project.
5. In the left navigation sidebar, choose Resources > **Machine Groups**.
6. On the right of **Machine Groups**, choose > **Create Machine Group**.
7. In the **Create Machine Group** dialog box, set the required parameters, and then click **OK**. The

following table describes the parameters.

Parameter	Description
Name	The name of the machine group. The name must be 2 to 128 characters in length, and can contain only lowercase letters, digits, hyphens (-), and underscores (_). The name must start and end with a lowercase letter or digit. **Notice** After the machine group is created, you cannot modify its name. Proceed with caution.
Identifier	The identifier of the server. Select **Custom ID**.
Topic	The topic of the machine group. This topic is used to differentiate log data that is generated in different servers. For more information, see Log topics.
Custom Identifier	Enter the custom ID that is set in .

8. View the status of the machine group.
   i. In the list of machine groups, click the destination machine group.
   ii. On the **Machine Group Settings** page, view the status of the machine group. You can view the list of servers that share the same custom ID. You can also view the heartbeat statuses of these servers.
   - The Machine Group Status section lists the IP addresses of the servers whose custom ID is the same as the custom ID that you set for the machine group.

     For example, the custom ID of a machine group is userdefined and the IP addresses in the Machine Group Status section are 10.10.10.10, 10.10.10.11, and 10.10.10.12. This indicates that you have specified the same custom ID for the servers in this machine group. If you want to add another server to the machine group and the IP address of the server is 10.10.10.13, set the custom ID to userdefined for the server. Then, you can view the IP address of the added server in the Machine Group Status section.
   - If the **Heartbeat** status is **OK**, the server is connected to Log Service. If the status is **FAIL**, see What can I do if the Logtail client has no heartbeat?



Disable a custom ID

If you want to set the Identifier parameter to IP Addresses, delete the user_defined_id file. The new configurations take effect within 1 minute.

- In Linux, run the following command:

```
rm -f /etc/ilogtail/user_defined_id
```

- In Windows, run the following command:

```
del C:\LogtailData\user_defined_id
```

Time to take effect

By default, after you add, delete, or modify the user_defined_id file, the new configurations take effect within 1 minute. If you want the configurations to immediately take effect, run the following command to restart Logtail.

- Linux:

```
/etc/init.d/ilogtaild stop
/etc/init.d/ilogtaild start
```

- Windows:
  i. Choose **Start Menu > Control Panel > Administrative Tools > Services**.
  ii. In the **Services** window, select the required service.
    - For Logtail V0.x.x.x, select LogtailWorker.
    - For Logtail V1.0.0.0 or later, select LogtailDaemon.
  iii. Right-click the service and then select **Restart** to validate the configurations.

3.4.5. Manage machine groups

This topic describes how to view the list of machine groups, modify a machine group, view the status of a machine group, manage Logtail configurations for a machine group, and delete a machine group in the Log Service console.

Prerequisites

Machine groups are created. For more information, see Create an IP address-based machine group or Create a custom ID-based machine group.

View the list of machine groups

1.
2.
3. In the left-side navigation pane, choose **Resources > Machine Groups**.
4. View the list of machine groups.

Modify a machine group

1. In the Machine Groups list, click the machine group that you want to modify.
2. On the **Machine Group Settings** page, click **Modify**.
3. Modify the configurations of the machine group and click **Save**.

View the status of a machine group

You can view the heartbeat status in a machine group to check whether a machine in the group is connected to Log Service.

> **Note** After you create a machine group, wait for 2 minutes to view the heartbeat status.

1. In the Machine Groups list, click the machine group whose status you want to view.
2. On the **Machine Group Settings** page, you can view the status of the machine group.
   - If a value in the Heartbeat column is OK, the machine is connected to Log Service.
   - If a value in the Heartbeat column is FAIL, the connection between the machine and Log Service is abnormal. Follow the on-screen instructions to troubleshoot the issue. For more information, see What do I do if a Logtail machine group has no heartbeats?. If the issue persists after troubleshooting, submit a ticket.

Manage Logtail configurations

You can create Logtail configurations in Log Service and apply the Logtail configurations to a machine group.

1. In the Machine Groups list, click the machine group whose Logtail configurations you want to manage.
2. On the **Machine Group Settings** page, click **Modify**.
3. In the **Configurations** section, add or remove Logtail configurations for the machine group. Then, click **Save**.

   After you add a Logtail configuration to the machine group, the Logtail configuration is applied to the Logtail that is installed on each server in the machine group. After you remove a Logtail configuration from the machine group, the Logtail configuration is no longer applied to the Logtail that is installed on each server in the machine group.

Delete a machine group

1. In the Machine Groups list, click the icon next to the machine group that you want to delete and select **Delete**.
2. In the message that appears, click **OK**.

3.4.6. Manage Logtail configurations for log collection

This topic describes how to create, view, modify, or delete Logtail configurations for log collection in the Log Service console.

Create Logtail configurations

For more information about how to create Logtail configurations in the Log Service console, see Collect text logs.

View Logtail configurations

1.
2. In the Projects section, click the target project.

3. Choose **Log Storage > Logstores**. Click the > icon of the target Logtail configurations. Choose **Data Import > Logtail Configurations**.
4. Click the target Logtail configurations to view the details of the configurations.

Modify Logtail configurations

1.
2. In the Projects section, click the target project.
3. Choose **Log Storage > Logstores**. Click the > icon of the target Logtail configurations. Choose **Data Import > Logtail Configurations**.
4. In the **Logtail Configurations** list, click the target Logtail configurations.
5. On the **Logtail Configurations** page, click **Modify**.
6. Modify the configurations based on your business requirements and then click **Save**.

   For more information about the parameters of the configurations, see Collect text logs.

Delete Logtail configurations

1. In the **Logtail Configurations** list, find the target Logtail configurations. Click the icon on the right of the target Logtail configurations, and then select **Delete**.
2. In the dialog box that appears, click **OK**.

   After the target Logtail configurations are deleted, the configurations are disassociated from the relevant machine group. Logtail no longer collects logs based on the configurations.

> **Note** Before you can delete a Logstore, you must delete all Logtail configurations that are associated with the Logstore.

3.5. Text logs

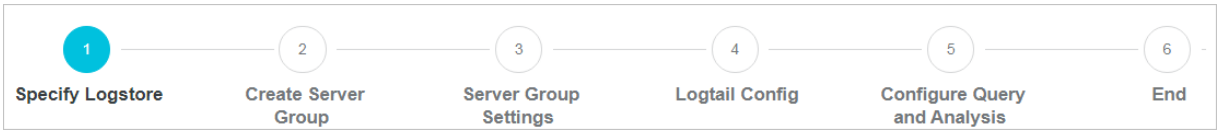
3.5.1. Overview

This topic describes the configuration process and collection modes when you use Logtail to collect text logs from servers.

Configuration process

Log Service provides a configuration wizard that you can use to configure log collection.

> **Note** Before you create a Logtail configuration, we recommend that you take note of the limits of Logtail. For more information, see Logtail limits.
>
> If the default settings of Logtail do not meet your collection requirements, you can modify the startup parameters of Logtail. For more information, see Configure the startup parameters of Logtail.



Collection modes

Logtail allows you to collect text logs in various modes. The following modes are supported: simple mode, full regex mode, delimiter mode, JSON mode, NGINX mode, Internet Information Services (IIS) mode, and Apache mode.

- Collect logs in simple mode
- Collect logs in full regex mode
- Collect logs in delimiter mode
- Collect logs in JSON mode
- Collect logs in NGINX mode
- Collect logs in IIS mode
- Collect logs in Apache mode

3.5.2. Collect logs in simple mode

When you collect logs in simple mode, the logs are not parsed. Each log is collected and then uploaded to Log Service as one log. This way, the process of log collection is simplified. This topic describes how to create a Logtail configuration in simple mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.

Context

In simple mode, you can collect the following types of text logs:

- Single-line text log

  Each log contains one line. Logs in a log file are separated by line feeds. If you collect single-line text logs, you need only to specify the directory and name of log files. This way, Logtail can collect the logs by line from the files that match the specified directory and name.

- Multi-line text log

  Each log contains multiple lines. If you collect multi-line text logs, you must specify the directory and name of log files. You must also enter a sample log and configure a regular expression to match the beginning of the first line of a log. Logtail uses the regular expression to match the beginning of the first line of a log and considers the content that does not match the regular expression as part of the log.

> **Note** If you collect logs in simple mode, the timestamp of a log is the system time when the log is collected. The system time refers to the time of the server on which Logtail runs.

Procedure

1.
2. In the **Import Data** section, click **Multi-line - Text Log**.

   This example shows how to collect multi-line text logs. If you want to collect single-line text logs,

click **Single Line - Text Log**.

3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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**.

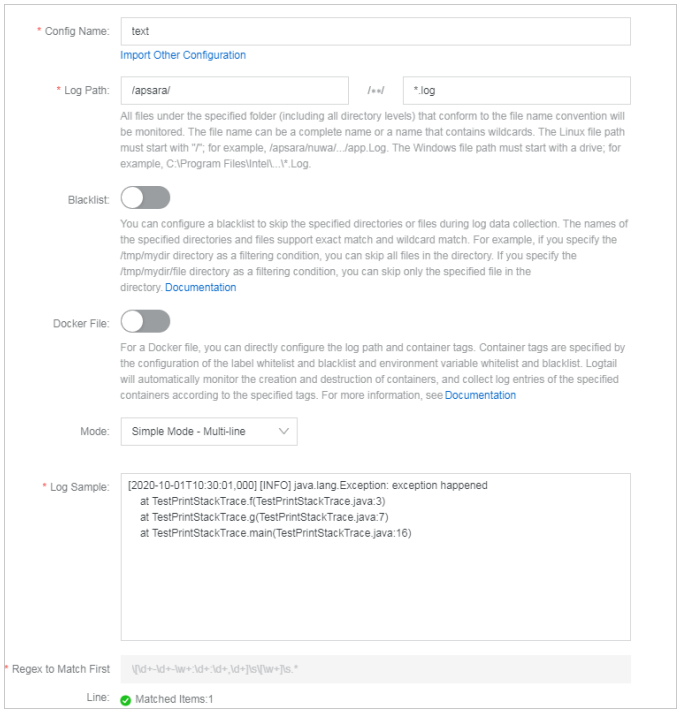
        For more information, see Install Logtail on ECS instances.

        > **Note** If you want to collect logs from an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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. Select the new machine group from **Source Server Groups** and move the machine group to **Applied Server Groups**. Then, click **Next**.

   > **Notice** If you apply a machine group immediately after you create the machine group, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. To resolve this issue, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
6. Create a Logtail configuration and click **Next**.



Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After you create the Logtail configuration, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import an existing Logtail configuration.

Parameter	Description
Log Path	Specify the directory and name of log files. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcard characters to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors the containers when the containers are created and destroyed, filters the logs of the containers by tag, and collects the filtered logs. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **Simple Mode - Multi-line** is displayed. You can change the mode.

Log Sample	Enter a sample log that is collected from an actual scenario. Then, Log Service can automatically generate a regular expression to match the beginning of the first line of the log. Examples:
	`[2020-10-01T10:30:01,000] [INFO] java.lang.Exception: exception happened`
	`at TestPrintStackTrace.f(TestPrintStackTrace.java:3)`
	`at TestPrintStackTrace.g(TestPrintStackTrace.java:7)`
	`at TestPrintStackTrace.main(TestPrintStackTrace.java:16)`
	If you want to collect single-line text logs in simple mode, you do not need to configure this parameter.

Regex to Match First Line	Configure a regular expression to match the beginning of the first line of a log. Logtail uses the regular expression to match the beginning of the first line of a log and considers the content that does not match the regular expression as part of the log. Log Service can automatically generate a regular expression or use the regular expression that you manually specify. ○ Automatic generation After you enter a sample multi-line text log, click **Auto Generate**. Log Service automatically generates a regular expression to match the beginning of the first line of the log. ○ Manual configuration After you enter a sample multi-line text log, click **Manual** and manually specify a regular expression to match the beginning of the first line of the log. Then, click **Validate** to check whether the regular expression is valid. For more information, see How do I modify a regular expression? If you want to collect single-line text logs in simple mode, you do not need to configure this parameter.
Drop Failed to Parse Logs	Specify whether to drop the logs that fail to be parsed. ○ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ○ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.

Parameter	Description
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. A value of 0 specifies that only the log file directory that you specify is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ◦ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ◦ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ◦ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ◦ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ◦ Custom: If you select this value, you must select a time zone based on your business requirements.

Parameter	Description
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.

More Configurations	Specify extended settings for Logtail. For more information, see advanced.
	For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail.
	`{`
	`"force_multiconfig": true,`
	`"batch_send_interval": 3`
	`}`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

3.5.3. Collect logs in full regex mode

You can use the full regex mode to extract custom fields from logs. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in full regex mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.

Procedure

1.
2. In the **Import Data** section, click **RegEx - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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. Create a Logtail configuration and click **Next**.

Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After you create the Logtail configuration, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import an existing Logtail configuration.
Log Path	Specify the directory and name of log files. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcard characters to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors containers to check whether containers are created or destroyed, filters containers by tag, and collects logs from the containers in the filtering result. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **Full Regex Mode** is displayed. You can change the mode.
Singleline	◦ If you want to collect single-line logs, turn on **Singleline**. Then, Log Service collects logs by line. ◦ If you want to collect multi-line logs such as Java program logs, turn off **Singleline**. Then, Log Service collects multi-line logs.
Log Sample	Enter a sample log that is obtained from an actual scenario. This way, Log Service can extract a regular expression from the log. For more information about sample logs, see Case: Collect single-line logs and Case: Collect multi-line logs.
Regex to Match First Line	Configure a regular expression to match the beginning of the first line of a log. If you want to collect multi-line logs, you must turn off **Singleline** and configure this parameter. Log Service can automatically generate a regular expression or use the regular expression that you manually specify. ◦ Automatic generation After you enter a sample multi-line log, click **Auto Generate**. Log Service automatically generates a regular expression to match the beginning of the first line of the log. ◦ Manual configuration After you enter a sample multi-line log, click **Manual** and manually specify a regular expression to match the beginning of the first line of the log. Then, click **Validate** to check whether the regular expression is valid. For more information, see How do I modify a regular expression?
Extract Field	If you turn on **Extract Field**, Log Service can extract key-value pairs by using a regular expression.

Parameter	Description
RegEx	If you turn on **Extract Field**, you must configure this parameter. ◦ Automatic generation In the **Log Sample** field, select the content that you want to extract and click **Generate Regular Expression**. A regular expression is automatically generated. ◦ Manual configuration Click **Manual** to specify a regular expression. Then, click **Validate** to check whether the regular expression can be used to parse logs or extract content from logs. For more information, see How do I modify a regular expression?
Extracted Content	If you turn on **Extract Field**, you must configure this parameter. After log content is extracted as values by using the regular expression, you must specify a key for each value.
Use System Time	If you turn on **Extract Field**, you must configure this parameter. ◦ If you turn on **Use System Time**, the timestamp of a log indicates the system time when the log is collected. The system time refers to the time of the server or container on which Logtail runs. ◦ If you turn off **Use System Time**, you must specify the time field for the **Extracted Content** parameter and configure **Time Conversion Format** based on the value of the time field. For more information about the time format, see Time formats. **Notice** ◦ The time zone of the Logtail container is UTC. If you want to collect container logs in DaemonSet mode and the time zone of the container from which you want to collect logs is not UTC, you must set **Timezone** to **Custom** in the **Advanced Options** section of your Logtail configuration and use the time zone of the container from which you want to collect logs. Otherwise, the log time is incorrectly offset. For example, if you select **Synchronize Timezone from Node to Container** when you create a container from which you want to collect logs, the time zone of the container may not be UTC. ◦ The timestamp of a log in Log Service is accurate to the second by default. If the value of the time field in raw logs has a higher time precision, such as the millisecond, microsecond, or nanosecond, and you want to retain the time precision for the logs in Log Service, you can add the enable_precise_timestamp parameter in the extended settings for your Logtail and set the parameter value to true.

Parameter	Description
Drop Failed to Parse Logs	Specify whether to drop the logs that fail to be parsed. ◦ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ◦ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. A value of 0 specifies that only the log file directory that you specify is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ◦ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ◦ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ◦ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.

Parameter	Description
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.

Parameter	Description
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

Case: Collect single-line logs

- Sample log

```
127.0.0.1 - - [10/Sep/2018:12:36:49  +0800] "GET /index.html HTTP/1.1" 200 612 "-" "Mozil
la/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
68.0.3440.106 Safari/537.36"
```

- Regular expression

```
(\S+)\s-\s(\S+)\s\[([^]]+)]\s"(\w+)\s(\S+)\s([^"]+)"\s(\d+)(\s\d+)[^-]+([^"]+)"\s"([^"]+)
.*
```

Case: Collect multi-line logs

- Sample log

```
[2018-10-01T10:30:01,000] [INFO] java.lang.Exception: exception happened
    at TestPrintStackTrace.f(TestPrintStackTrace.java:3)
    at TestPrintStackTrace.g(TestPrintStackTrace.java:7)
    at TestPrintStackTrace.main(TestPrintStackTrace.java:16)
```

- Regular expression that is used to match the beginning of the first line of the log

```
\[\d+-\d+-\w+:\d+:\d+,\d+]\s\[\w+]\s.*
```

- Regular expression

```
\[(\S+)]\s\[(\S+)]\s(.*)
```

3.5.4. Collect logs in NGINX mode

Log Service allows you to collect NGINX logs and analyze logs in multiple dimensions. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in NGINX mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.

Procedure

1.
2. In the **Import Data** section, click **Nginx - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

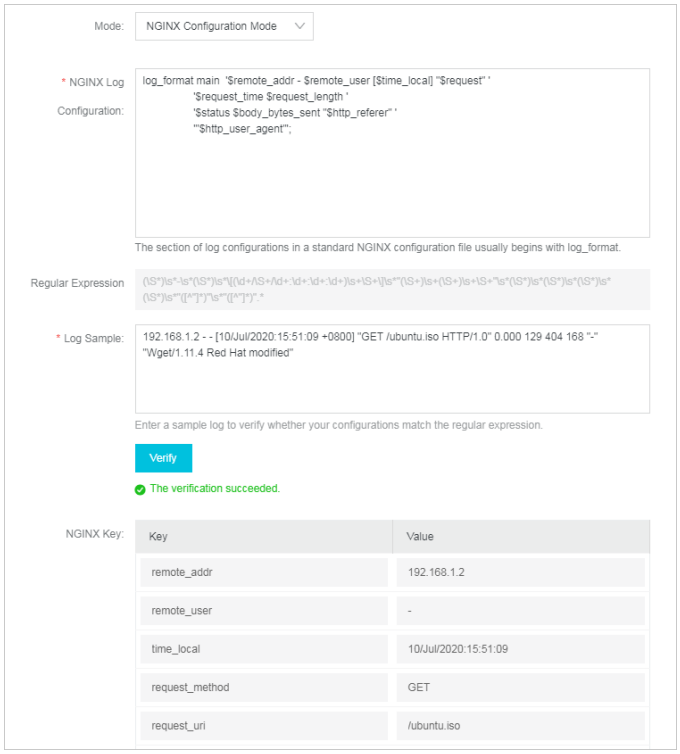
     b. After Logtail is installed, click **Complete Installation**.

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. Create a Logtail configuration and click **Next**.



Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After the Logtail configuration is created, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import a Logtail configuration from another project.

Parameter	Description
Log Path	Specify the log file directory and log file name. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcards to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors the containers to check whether the containers are created or destroyed. Then, Logtail filters the logs of the containers based on tags and collects the logs that meet the filter conditions. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **NGINX Configuration Mode** is displayed. You can change the mode.

NGINX Log Configuration	Enter the log configuration section that is specified in the NGINX configuration file. The section starts with **log_format**. Example:
	`log_format main '$remote_addr - $remote_user [$time_local] "$request" '`
	`'$request_time $request_length '`
	`'$status $body_bytes_sent "$http_referer" '`
	`'"$http_user_agent"';`
	For more information, see Additional information: Log formats and sample logs.

Regular Expression	Log Service generates a regular expression based on the content that you enter in **NGINX Log Configuration**.
Log Sample	Enter a sample NGINX log that is collected from an actual scenario. Example: `192.168.1.2 - - [10/Jul/2020:15:51:09 +0800] "GET /ubuntu.iso HTTP/1.0" 0.000 129 404 168 "-" "Wget/1.11.4 Red Hat modified"` Log Service uses the sample log to check whether the content of **NGINX Log Configuration** matches the generated regular expression. After you enter the sample log, click **Verify**. If the verification is successful, Log Service automatically extracts the values for **NGINX Key** from the sample log.
NGINX Key	The NGINX keys and values are automatically generated based on the content of NGINX Log Configuration and Log Sample.
Drop Failed to Parse Logs	◦ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ◦ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. The value 0 indicates that only the specified log file directory is monitored.

Parameter	Description

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ○ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ○ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ○ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.

Parameter	Description
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.

More Configurations	Specify extended settings for Logtail. For more information, see advanced.
	For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail.
	`{`
	`"force_multiconfig": true,`
	`"batch_send_interval": 3`
	`}`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

Additional information: Log formats and sample logs

Before you collect NGINX logs, you must configure log_format and access_log in the */etc/nginx/nginx.conf* file. log_format specifies the log format, and access_log specifies the directory in which the NGINX log files are stored.

- Log format

  In the following example, the default values of log_format and access_log are used:

```
log_format main  '$remote_addr - $remote_user [$time_local] "$request" '
                 '$request_time $request_length '
                 '$status $body_bytes_sent "$http_referer" '
                 '"$http_user_agent"';
access_log /var/logs/nginx/access.log main
```

  The following table describes the log fields.

Field	Description
remote_addr	The IP address of the client.
remote_user	The username that is used by the client to send a request.
time_local	The system time of the server. The value must be enclosed in brackets [].
request	The URL and HTTP protocol of a request.
request_time	The time that is required to process a request. Unit: seconds.
request_length	The length of a request. The request line, request headers, and request body are all counted.
status	The status of a request.
body_bytes_sent	The number of bytes in a response that is sent to the client. The response header is not counted.
http_referer	The URL of the source web page.

Field	Description
http_user_agent	The browser information of the client.

- Sample log

```
192.168.1.2 - - [10/Jul/2020:15:51:09 +0800] "GET /ubuntu.iso HTTP/1.0" 0.000 129 404 168
"-" "Wget/1.11.4 Red Hat modified"
```

3.5.5. Collect logs in delimiter mode

Log Service allows you to collect logs in delimiter mode. After logs are collected, you can perform various operations on the logs. For example, you can analyze the logs in multiple dimensions, and transform and ship the logs. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in delimiter mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.

Procedure

1.
2. In the **Import Data** section, click **Delimiter Mode - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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. Create a Logtail configuration and click **Next**.

Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After you create the Logtail configuration, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import an existing Logtail configuration.
Log Path	Specify the directory and name of log files. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors containers to check whether containers are created or destroyed, filters containers by tag, and collects logs from the containers in the filtering result. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcard characters to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.
Mode	Select the log collection mode. By default, **Delimiter Mode** is displayed. You can change the mode.

Parameter	Description
Log Sample	Enter a sample log that is obtained from an actual scenario. Example: `127.0.0.1|#|-|#|13/Apr/2020:09:44:41 +0800|#|GET /1 HTTP/1.1|#|0.000|#|74|#|404|#|3650|#|-|#|curl/7.29.0` **Note** In delimiter mode, you can collect only single-line logs. If you want to collect multi-line logs, we recommend that you select Simple Mode - Multi-line or Full Regex Mode. For more information, see Collect logs in simple mode and Collect logs in full regex mode.
Delimiter	Select a delimiter based on the log format. For example, you can select a vertical bar (|) as a delimiter. For more information, see Additional information: Delimiters and sample logs. **Note** If you select **Hidden Characters** for Delimiter, you must enter a character in the following format: `0x`*Hexadecimal ASCII code of the non-printable character*. For example, if you want to use the non-printable character whose hexadecimal ASCII code is 01, you must enter **0x01**.
Quote	Select a quote to enclose the log fields that contain delimiters. Log Service parses the content that is enclosed in a pair of quotes into a complete field. You can select the quote based on the log format. **Note** If you select **Hidden Characters** for Quote, you must enter a character in the following format: `0x`*Hexadecimal ASCII code of the non-printable character*. For example, if you want to use the non-printable character whose hexadecimal ASCII code is 01, you must enter **0x01**.
Extracted Content	Specify the log content that can be extracted. Log Service extracts log content based on the sample log that you enter, and then delimits the log content into values by using the specified delimiter. You must specify a key for each value.
Incomplete Entry Upload	Specify whether to upload a log if the number of fields parsed from the log is less than the number of specified keys. If you turn on this switch, the log is uploaded. Otherwise, the log is discarded. For example, if you specify a vertical bar (|) as the delimiter, the log 11|22|33|44|55 is parsed into the following fields: 11, 22, 33, 44, and 55. You can set the keys to A, B, C, D, and E. ◦ If you turn on **Incomplete Entry Upload**, the log 11|22|33|55 is uploaded, and 55 is uploaded as the value of the D key. ◦ If you turn off **Incomplete Entry Upload**, the log 11|22|33|55 is discarded because the number of fields parsed from the log does not match the number of specified keys.

Parameter	Description
Use System Time	Specify whether to use the system time. ◦ If you turn on **Use System Time**, the timestamp of a log indicates the system time when the log is collected. The system time refers to the time of the server or container on which Logtail runs. ◦ If you turn off **Use System Time**, you must configure **Specify Time Key** and **Time Format** based on the value of the time field in logs. For more information about the time format, see Time formats. For example, if the time format in logs is **"time": "05/May/2016:13:30:28"**, you can set the **Specify Time Key** field to **time** and the **Time Format** field to **%d/%b/%Y:%H:%M:%S**. **Notice** ◦ The time zone of the Logtail container is UTC. If you want to collect container logs in DaemonSet mode and the time zone of the container from which you want to collect logs is not UTC, you must set **Timezone** to **Custom** in the **Advanced Options** section of your Logtail configuration and use the time zone of the container from which you want to collect logs. Otherwise, the log time is incorrectly offset. For example, if you select **Synchronize Timezone from Node to Container** when you create a container from which you want to collect logs, the time zone of the container may not be UTC. ◦ The timestamp of a log in Log Service is accurate to the second by default. If the value of the time field in raw logs has a higher time precision, such as the millisecond, microsecond, or nanosecond, and you want to retain the time precision for the logs in Log Service, you can add the enable_precise_timestamp parameter in the extended settings for your Logtail and set the parameter value to true.
Drop Failed to Parse Logs	Specify whether to drop the logs that fail to be parsed. ◦ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ◦ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. A value of 0 specifies that only the log file directory that you specify is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ○ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ○ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ○ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.

Parameter	Description
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based

on collected logs in manual or automatic mode. For more information, see Configure indexes.

> **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

Additional information: Delimiters and sample logs

Logs that are in the delimiter-separated values (DSV) format use line feeds as boundaries. Each log is placed in a separate line. Each log is parsed into multiple fields by using delimiters. Both single-character and multi-character delimiters are supported. If a field contains delimiters, you can enclose the field in a pair of quotes.

- Single-character delimiter

  The following examples show logs that use single-character delimiters:

```
05/May/2016:13:30:28,10.10.*.*,"POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=****
************&Date=Fri%2C%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=*****
************************** HTTP/1.1",200,18204,aliyun-sdk-java
05/May/2016:13:31:23,10.10.*.*,"POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=****
************&Date=Fri%2C%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=*****
************************** HTTP/1.1",401,23472,aliyun-sdk-java
```

  If a log uses single-character delimiters, you must specify the delimiter. You can also specify a quote.

  - Delimiter: Available single-character delimiters include the tab character (\t), vertical bar (|), space, comma (,), semicolon (;), and non-printable characters. You cannot specify a double quotation mark (") as the delimiter.

    However, a double quotation mark (") can be used as a quote. A double quotation mark (") can appear at the border of a field, or in the field. If a double quotation mark (") is included in a log field, it must be escaped as a pair of double quotation marks ("") when the log is processed. When the log is parsed, a pair of double quotation marks ("") are restored to a double quotation mark ("). For example, you can specify a comma (,) as the delimiter and a double quotation mark (") as the quote. If a log field contains the specified delimiter and quote, the field is enclosed in a pair of quotes, and the double quotation mark (") in the field is escaped as a pair of double quotation marks (""). If a processed log is in the format 1999,Chevy,"Venture ""Extended Edition, Very Large""","",5000.00, the log is parsed into five fields: 1999, Chevy, Venture "Extended Edition, Very Large", an empty field, and 5000.00.

  - Quote: If a log field contains delimiters, you must specify a quote to enclose the field. Log Service parses the content that is enclosed in a pair of quotes into a complete field.

    Available quotes include the tab character (\t), vertical bar (|), space, comma (,), semicolon (;), and non-printable characters.

    For example, if you specify a comma (,) as the delimiter and a double quotation mark (") as the quote, the log 1997,Ford,E350,"ac, abs, moon",3000.00 is parsed into five fields: 1997, Ford, E350, ac, abs, moon, and 3000.00.

- Multi-character delimiter

  The following examples show logs that use multi-character delimiters:

```
05/May/2016:13:30:28&&10.200.**.**&&POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=
***************&Date=Fri%2C%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=p
D12XYLmGxKQ%2Bmkd6x7hAgQ7b1c%3D HTTP/1.1&&200&&18204&&aliyun-sdk-java
05/May/2016:13:31:23&&10.200.**.**&&POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=
***************&Date=Fri%2C%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=*
****************************** HTTP/1.1&&401&&23472&&aliyun-sdk-java
```

A multi-character delimiter can contain two or three characters, such as ||, &&&, and ^_^. Log Service parses logs based on delimiters. You do not need to use quotes to enclose log fields.

> **Note** Make sure that each log field does not contain the exact delimiter. Otherwise, Log Service cannot parse the logs as expected.

For example, if you specify && as the delimiter, the log 1997&&Ford&&E350&&ac&abs&moon&&3000.00 is parsed into five fields: 1997, Ford, E350, ac&abs&moon, and 3000.00.

3.5.6. Collect logs in JSON mode

Log Service allows you to collect JSON logs in JSON mode by using Logtail. After logs are collected, you can perform various operations on the logs. For example, you can analyze the logs in multiple dimensions, and transform and ship the logs. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in JSON mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.

Context

JSON logs can be written in the object or array structure. A log in the object structure contains key-value pairs, and a log in the array structure contains an ordered list of values.

In JSON mode, Logtail can parse JSON logs in the object structure and extract the keys and values from the first layer of each object. The extracted keys are used as field names, and the extracted values are used as field values. Logtail cannot parse JSON logs in the array structure. In this case, you can collect data from the JSON logs in full regex or simple mode. For more information, see Collect logs in simple mode or Collect logs in full regex mode.

Sample JSON logs:

```
{"url": "POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=U0Ujpek********&Date=Fri%2C%2
028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=pD12XYLmGxKQ%2Bmkd6x7hAgQ7b1c%3D
HTTP/1.1", "ip": "10.200.98.220", "user-agent": "aliyun-sdk-java", "request": {"status": "2
00", "latency": "18204"}, "time": "05/Jan/2020:13:30:28"}
{"url": "POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=U0Ujpek********&Date=Fri%2C%2
028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=pD12XYLmGxKQ%2Bmkd6x7hAgQ7b1c%3D
HTTP/1.1", "ip": "10.200.98.210", "user-agent": "aliyun-sdk-java", "request": {"status": "2
00", "latency": "10204"}, "time": "05/Jan/2020:13:30:29"}
```

Procedure

1.
2. In the **Import Data** section, click **JSON - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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. Create a Logtail configuration and click **Next**.

Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After you create the Logtail configuration, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import an existing Logtail configuration.

Parameter	Description
Log Path	Specify the directory and name of log files. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcard characters to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors containers to check whether containers are created or destroyed, filters containers by tag, and collects logs from the containers in the filtering result. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **JSON Mode** is displayed. You can change the mode.
Use System Time	Specify whether to use the system time. ◦ If you turn on **Use System Time**, the timestamp of a log indicates the system time when the log is collected. The system time refers to the time of the server or container on which Logtail runs. ◦ If you turn off **Use System Time**, you must configure **Specify Time Key** and **Time Format** based on the value of the time field in logs. For more information about the time format, see Time formats. For example, if the time format in logs is **"time": "05/May/2016:13:30:28"**, you can set the **Specify Time Key** field to **time** and the **Time Format** field to **%d/%b/%Y:%H:%M:%S**. **Notice** ◦ The time zone of the Logtail container is UTC. If you want to collect container logs in DaemonSet mode and the time zone of the container from which you want to collect logs is not UTC, you must set **Timezone** to **Custom** in the **Advanced Options** section of your Logtail configuration and use the time zone of the container from which you want to collect logs. Otherwise, the log time is incorrectly offset. For example, if you select **Synchronize Timezone from Node to Container** when you create a container from which you want to collect logs, the time zone of the container may not be UTC. ◦ The timestamp of a log in Log Service is accurate to the second by default. If the value of the time field in raw logs has a higher time precision, such as the millisecond, microsecond, or nanosecond, and you want to retain the time precision for the logs in Log Service, you can add the enable_precise_timestamp parameter in the extended settings for your Logtail and set the parameter value to true.
Drop Failed to Parse Logs	Specify whether to drop the logs that fail to be parsed. ◦ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ◦ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.

Parameter	Description
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. A value of 0 specifies that only the log file directory that you specify is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ○ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ○ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ○ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.

Parameter	Description
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.

More Configurations	Specify extended settings for Logtail. For more information, see advanced.
	For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail.
	`{`
	`"force_multiconfig": true,`
	`"batch_send_interval": 3`
	`}`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

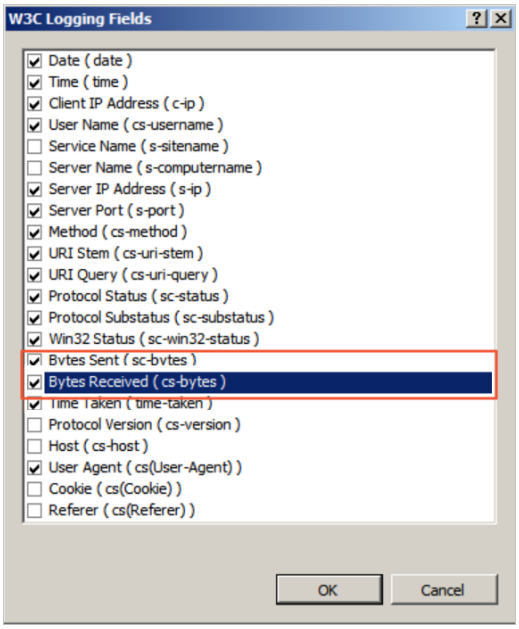
3.5.7. Collect logs in IIS mode

Log Service allows you to collect Internet Information Services (IIS) logs and analyze logs in multiple dimensions. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in IIS mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.
- Logs are generated on the server in the IIS, NCSA Common, or W3C Extended format.

  We recommend that you use the W3C Extended log format. If you select the W3C Extended format, you must configure log fields beforehand. To configure log fields, you must select **Bytes Sent (sc-bytes)** and **Bytes Received (cs-bytes)** in the W3C Logging Fields dialog box and retain the default settings for other fields.



Procedure

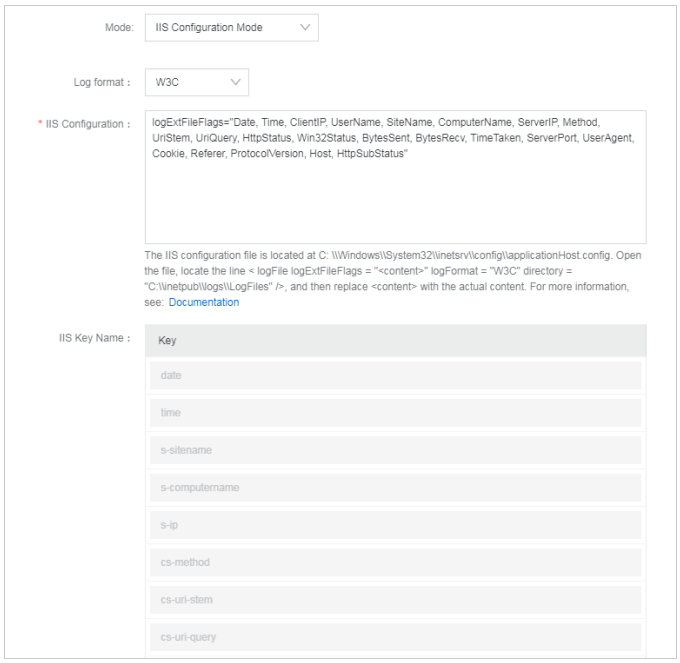
1.
2. In the **Import Data** section, click **IIS - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use Existing Machine Groups**.
   - If no machine groups are available, perform the following steps to create 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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 **Logtail Config** step, create a Logtail configuration.



Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After the Logtail configuration is created, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import a Logtail configuration from another project.
Log Path	Specify the log file directory and log file name. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcards to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors the containers to check whether the containers are created or destroyed. Then, Logtail filters the logs of the containers based on tags and collects the logs that meet the filter conditions. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **IIS Configuration Mode** is displayed. You can change the mode. For more information about other modes, see Overview.
Log format	Select the format of logs that are generated on the IIS server. ○ IIS: Microsoft IIS log file format ○ NCSA: NCSA Common log file format ○ W3C: W3C Extended log file format
IIS Configuration	Specify the IIS configuration fields. ○ If you select IIS or NCSA for Log Format, the IIS configuration fields are automatically set. ○ If you select W3C for Log Format, enter the content that is specified for the logFile logExtFileFlags parameter of the IIS configuration file. `logExtFileFlags="Date, Time, ClientIP, UserName, SiteName, ComputerName, ServerIP, Method, UriStem, UriQuery, HttpStatus, Win32Status, BytesSent, BytesRecv, TimeTaken, ServerPort, UserAgent, Cookie, Referer, ProtocolVersion, Host, HttpSubStatus"` ■ Default path of the IIS5 configuration file: C:\WINNT\system32\inetsrv\MetaBase.bin ■ Default path of the IIS6 configuration file: C:\WINDOWS\system32\inetsrv\MetaBase.xml ■ Default path of the IIS7 configuration file: C:\Windows\System32\inetsrv\config\applicationHost.config
IIS Key Name	Log Service automatically extracts IIS keys based on the content of **IIS Configuration**.
Drop Failed to Parse Logs	○ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ○ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.

Parameter	Description
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. The value 0 indicates that only the specified log file directory is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ◦ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ◦ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ◦ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ◦ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ◦ Custom: If you select this value, you must select a time zone based on your business requirements.

Parameter	Description
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

Additional information: Sample logs and field descriptions

The following example shows a sample IIS log:

```
#Software: Microsoft Internet Information Services 7.5
#Version: 1.0
#Date: 2020-09-08 09:30:26
#Fields: date time s-sitename s-ip cs-method cs-uri-stem cs-uri-query s-port cs-username c-ip cs(User-Agent) sc-status sc-substatus sc-win32-status sc-bytes cs-bytes time-taken
2009-11-26 06:14:21 W3SVC692644773 125.67.67.* GET /index.html - 80 - 10.10.10.10 Baiduspider+(+http://www.example.com)200 0 64 185173 296 0
```

- Field prefixes

Prefix	Description
s-	Indicates a server action.
c-	Indicates a client action.
cs-	Indicates a client-to-server action.
sc-	Indicates a server-to-client action.

- Fields

Field	Description
date	The date on which the client sends the request.
time	The point in time at which the client sends the request.
s-sitename	The Internet service name and instance ID of the site that is visited by the client.
s-computername	The name of the server on which the log is generated.
s-ip	The IP address of the server on which the log is generated.

Field	Description
cs-method	The request method that is used by the client, such as GET or POST.
cs-uri-stem	The URI in the request.
cs-uri-query	The query string that follows the question mark (?) in the HTTP request.
s-port	The port number of the server.
cs-username	The authenticated domain name or username that is used by the client to access the server. ○ Authenticated users are indicated in the `Domain\Username` format. ○ Anonymous users are indicated by a hyphen (-).
c-ip	The actual IP address of the client that sends the request.
cs-version	The protocol version that is used by the client, such as HTTP 1.0 or HTTP 1.1.
cs(User-Agent)	The browser used by the client.
Cookie	The content of the cookie that is sent or received. If no cookies are sent or received, a hyphen (-) is displayed.
referer	The site from which the client is directed.
cs-host	The host information.
sc-status	The HTTP status code returned by the server.
sc-substatus	The HTTP substatus code returned by the server.
sc-win32-status	The Windows status code returned by the server.
sc-bytes	The number of bytes sent by the server.
cs-bytes	The number of bytes received by the server.
time-taken	The time required to process the request. Unit: milliseconds.

3.5.8. Collect logs in Apache mode

Log Service allows you to collect Apache logs and analyze logs in multiple dimensions. Apache logs record O&M information about websites. You can create Logtail configurations to collect logs. This topic describes how to create a Logtail configuration in Apache mode by using the Log Service console.

Prerequisites

- A project and a Logstore are created. For more information, see Create a project and Create a

Logstore.

- The server on which Logtail is installed can connect to port 80 and port 443 of remote servers.
- The log display format, log file directory, and log file name are specified in the Apache configuration file. For more information, see Additional information: Log formats and sample logs.

Procedure

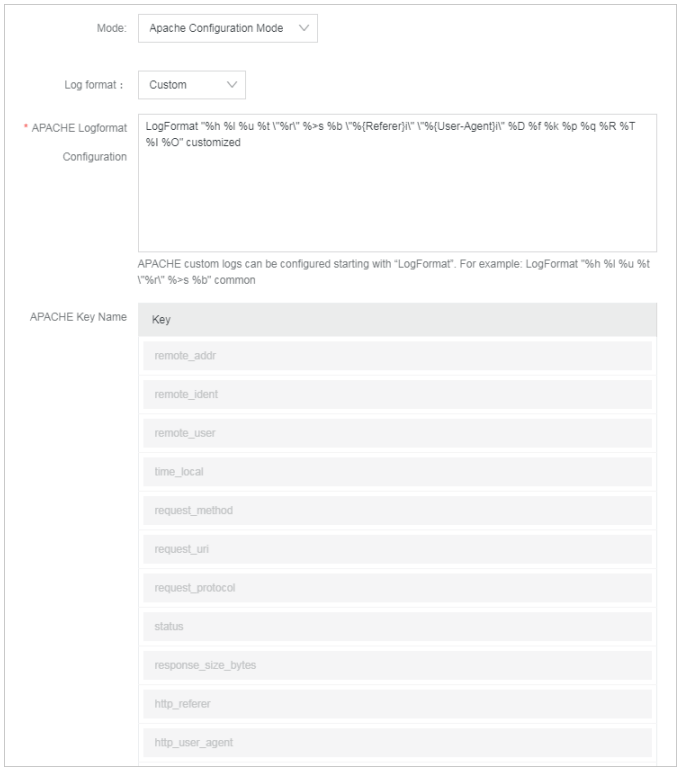
1.
2. In the **Import Data** section, click **Apache - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use Existing Machine Groups**.
   - If no machine groups are available, perform the following steps to 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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 **Logtail Config** step, create a Logtail configuration.



Parameter	Description
Config Name	Enter a name for the Logtail configuration. The name must be unique in a project. After the Logtail configuration is created, you cannot change the name of the Logtail configuration. You can click **Import Other Configuration** to import a Logtail configuration from another project.

Parameter	Description
Log Path	Specify the log file directory and log file name. You can specify an exact directory and an exact name. You can also use wildcard characters to specify the directory and name. For more information, see Wildcard matching. Log Service scans all levels of the specified directory for the log files that match specified conditions. Examples: ◦ If you specify */apsara/nuwa/**/*.log*, Log Service collects logs from the log files whose names are suffixed by *.log* in the */apsara/nuwa* directory and the recursive subdirectories of the directory. ◦ If you specify */var/logs/app_*/*.log*, Log Service collects logs from the log files that meet the following conditions: The file name contains *.log*. The file is stored in a subdirectory under the */var/logs* directory or in a recursive subdirectory of the subdirectory. The name of the subdirectory matches the *app_** pattern. **Note** ◦ By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?. ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters.

Parameter	Description
Blacklist	If you turn on **Blacklist**, you must configure a blacklist to specify the directories or files that you want Log Service to skip when it collects logs. You can specify exact directories and file names. You can also use wildcards to specify directories and file names. Examples: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped. **Note** ◦ When you configure this parameter, you can use only asterisks (*) or question marks (?) as wildcard characters. ◦ If you use wildcard characters to configure **Log Path** and you want to skip some directories in the specified directory, you must configure the blacklist and enter a complete directory. For example, if you set **Log Path** to */home/admin/app*/log/*.log* and you want to skip all subdirectories in the */home/admin/app1** directory, you must select **Filter by Directory** and enter */home/admin/app1*/**to configure the blacklist. If you enter */home/admin/app1**, the blacklist does not take effect. ◦ When a blacklist is in use, computational overhead is generated. We recommend that you add up to 10 entries to the blacklist.

Parameter	Description
Docker File	If you want to collect logs from Docker containers, you must turn on **Docker File** and specify the directories and tags of the containers. Logtail monitors the containers to check whether the containers are created or destroyed. Then, Logtail filters the logs of the containers based on tags and collects the logs that meet the filter conditions. For more information about how to collect the text logs of containers, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Select the log collection mode. By default, **Apache Configuration Mode** is displayed. You can change the mode.
Log format	Select the log format that is specified in the Apache configuration file. Valid values: common, combined, and Custom.
APACHE Logformat Configuration	Enter the log configuration section that is specified in the Apache configuration file. In most cases, the section starts with LogFormat. For more information, see Additional information: Log formats and sample logs. ◦ If you set **Log format** to **common** or **combined**, the system automatically inserts a value into this field. Check whether the value is the same as that specified in the Apache configuration file. ◦ If you set **Log format** to **Custom**, specify a value based on your business requirements. For example, you can enter **LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %D %f %k %p %q %R %T %I %O" customized**.
APACHE Key Name	Log Service automatically reads Apache keys from the value of the **APACHE Logformat Configuration** field.
Drop Failed to Parse Logs	Specify whether to drop the logs that fail to be parsed. ◦ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ◦ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	Specify the maximum number of levels of subdirectories that you want to monitor. The subdirectories are in the log file directory that you specify. Valid values: 0 to 1000. The value 0 indicates that only the specified log file directory is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ○ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ○ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ○ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.

Parameter	Description
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

Click **Next** to complete the Logtail configuration creation. Then, Log Service starts to collect logs.

- A Logtail configuration requires up to 3 minutes to take effect.
- If an error occurs when you use Logtail to collect logs, see How do I view Logtail collection errors?.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based

on collected logs in manual or automatic mode. For more information, see Configure indexes.

> **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

Additional information: Log formats and sample logs

Before you collect Apache logs, you must specify the log display format, log file directory, and log file name. For example, if you enter **CustomLog "/var/log/apache2/access_log" combined**, logs are displayed in the combined log format, and the log file directory is */var/log/apache2/access_log*. The following list describes supported log formats and provides a sample log:

- Log formats
  - The combined log format:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
```

  - The common log format:

```
LogFormat "%h %l %u %t \"%r\" %>s %b"
```

  - A custom log format:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %D %f %k %p %q
%R %T %I %O" customized
```

The following table describes the fields for these log formats. For more information, see mod_log_config.

Field	Key	Description
%a	client_addr	The IP address of the client.
%A	local_addr	The local IP address.
%b	response_size_bytes	The number of bytes in a response. If no bytes are sent, a hyphen (-) is displayed for this field.
%B	response_bytes	The number of bytes in a response. If no bytes are sent, the digit 0 is displayed for this field.
%D	request_time_msec	The time required to process a request. Unit: microseconds.
%f	filename	The file name.
%h	remote_addr	The name of the remote host.
%H	request_protocol_supple	The request protocol.

Field	Key	Description
%I	bytes_received	The number of bytes that are received by the server. This field is recorded in logs only after you enable the mod_logio module.
%k	keep_alive	The number of keep-alive requests handled on the connection.
%l	remote_ident	The information that is provided by the remote host for identification.
%m	request_method_supple	The HTTP request method.
%O	bytes_sent	The number of bytes that are sent by the server. This field is recorded in logs only after you enable the mod_logio module.
%p	remote_port	The port number of the server.
%P	child_process	The ID of the child process.
%q	request_query	The query string. If no query strings exist, an empty string is displayed.
%r	request	The first line of a request. This line consists of the HTTP request method, address, and HTTP version.
%R	response_handler	The type of the handler that generates a response on the server.
%s	status	The initial HTTP status of a response.
%>s	status	The final HTTP status of a response.
%t	time_local	The point in time at which the server receives a request.
%T	request_time_sec	The time required to process a request. Unit: seconds.
%u	remote_user	The username that is used by the client to send a request.
%U	request_uri_supple	The URI in a request. The URI does not include the query string.
%v	server_name	The name of the server.
%V	server_name_canonical	The name of the server. The name is specified by using the UseCanonicalName directive.

Field	Key	Description
"%{User-Agent}i"	http_user_agent	The information about the client.
"%{Rererer}i"	http_referer	The URL of the source web page.

- Sample log

```
192.168.1.2 - - [02/Feb/2020:17:44:13 +0800] "GET /favicon.ico HTTP/1.1" 404 209 "http://localhost/x1.html" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/48.0.2564.97 Safari/537.36"
```

3.5.9. Import historical logs

This topic describes how to import historical logs from a server to Log Service. By default, Logtail only collects incremental logs from servers. However, you can configure Logtail to collect historical logs.

Prerequisites

- Logtail V0.16.15 (Linux), Logtail V1.0.0.1 (Windows), or later is installed on the server. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.
- A Logtail configuration file is created and applied to the server group to which the server belongs. For more information, see Overview of text log collection.

  If the Logtail configuration file is used to only import historical files, you can specify a log collection path that does not exist.

Context

Logtail collects logs based on the monitoring of log file modifications. Logtail can also load events from local files to collect logs. Logtail collects historical logs by loading local events.

You must import historical log files in the directory where Logtail is installed. The directory varies depending on different operating systems.

- Linux: */usr/local/ilogtail*
- Windows:
  - 32-bit: *C:\Program Files\Alibaba\Logtail*
  - 64-bit: *C:\Program Files (x86)\Alibaba\Logtail*

> **Note**
> - The maximum latency that is allowed to import local events is 1 minute.
> - If a local event is loaded, Logtail sends the `LOAD_LOCAL_EVENT_ALARM` message to the server.
> - To import a large number of files, we recommend that you modify the Logtail startup parameters. You can set the threshold of the CPU usage to 2.0 or a larger value, and set the memory usage required by Logtail to 512 MB or a larger value. For more information, see Configure the startup parameters of Logtail.

Procedure

1. Obtain the unique identifier of the Logtail configuration file.

   Open the *user_log_config.json* file in the directory where Logtail is installed. You can obtain the unique identifier of the Logtail configuration file.

   For example, to obtain the unique identifier of Logtail configuration file in a Linux server, run the following command:

   ```
   grep "##" /usr/local/ilogtail/user_log_config.json | awk '{print $1}'
                       "##1.0##log-config-test$multi"
                       "##1.0##log-config-test$ecs-test"
                       "##1.0##log-config-test$metric_system_test"
                       "##1.0##log-config-test$redis-status"
   ```

2. Add a local event.

   i. Create the *local_event.json* file in the Logtail installation directory.

ii. Add the local event in the JSON format to the *local_event.json* file of the Logtail installation directory. The following example shows the format of the local event:

```
[
  {
    "config" : "${your_config_unique_id}",
    "dir" : "${your_log_dir}",
    "name" : "${your_log_file_name}"
   },
  {
   ...
   }
   ...
]
```

Note To prevent Logtail from loading invalid JSON files, we recommend that you first save the configurations of the local event to a temporary file. Then, edit and copy the configurations to the *local_event.json* file.

Parameter	Description
config	Enter the unique identifier that is obtained in Step 1. Example: ##1.0##log-config-test$ecs-test.
dir	The directory in which historical log files are saved. Example: */data/logs*. **Note** The directory cannot end with a forward slash (/).
name	The name of the historical log file. A name with wildcards is supported. Example: access.log.2018-08-08 and access.log*.

The following example describes how to configure a local event in Linux.

```
$ cat /usr/local/ilogtail/local_event.json
[
  {
    "config": "##1.0##log-config-test$ecs-test",
    "dir": "/data/log",
    "name": "access.log*"
  },
  {
    "config": "##1.0##log-config-test$tmp-test",
    "dir": "/tmp",
    "name": "access.log.2017-08-09"
  }
]
```

FAQ

- How do I check whether Logtail loads Logtail configurations?

  After you save the *local_event.json* file, Logtail loads the configurations of the local event to the memory within 1 minute. Then, the content of the *local_event.json* file is deleted.

  You can use the following methods to check whether the Logtail configurations are loaded.

  i. If no content exists in the *local_event.json* file, it indicates that the local event in the file is read by Logtail.

  ii. Check whether the *ilogtail.LOG* file in the Logtail installation directory contains the process local event parameter. If the content in the *local_event.json* file is cleared but the process local event parameter does not exist, the content of the *local_event.json* file may be invalid and filtered out.

- Why am I unable to collect a log file after Logtail configurations are loaded?
  - The Logtail configurations are invalid.
  - The configurations of the local event in the *local_event.json* file are invalid.
  - The log file does not exist in the path that is specified in the Logtail configurations.
  - The log file has been collected by Logtail.

3.5.10. Time formats

If you use Logtail to collect logs, you must specify time formats based on the time strings in raw logs. Logtail extracts a time string from a raw log and parses the string into a UNIX timestamp. This topic describes the time formats that are commonly used in logs and provides examples for the time formats.

Commonly used time formats in logs

The following table describes the time formats that are supported by Logtail.

> **Note**
>
> - The timestamp of a log in Log Service is accurate to the second by default. Therefore, you need to only specify a time format that is accurate to the second.
>
> If the value of the time field in raw logs has a higher time precision, such as the millisecond, microsecond, or nanosecond, and you want to retain the time precision for the logs in Log Service, you can add the enable_precise_timestamp parameter in the extended settings for your Logtail and set the parameter value to true. For more information, see Advanced settings and Parameters of advanced.
>
> - You need to only specify a time format for the time part in a time string. You do not need to specify a time format for other parts such as a time zone.
> - If Logtail is installed on a Linux server, Logtail supports all the time formats that are supported by the strftime function. If the time string in a log can be formatted by the strftime function, the time string can be parsed and used by Logtail.

Time format	Description	Example
%a	The abbreviated name of the day of the week.	Fri

Time format	Description	Example
%A	The full name of the day of the week.	Friday
%b	The abbreviated name of the month.	Jan
%B	The full name of the month.	January
%d	The day of the month. The value is in the decimal format. Valid values: 01 to 31.	07, 31
%h	The abbreviated name of the month. The format is equivalent to %b.	Jan
%H	The hour. The 24-hour clock is used.	22
%I	The hour. The 12-hour clock is used.	11
%m	The month. The value is in the decimal format. Valid values: 01 to 12.	08
%M	The minute. The value is in the decimal format. Valid values: 00 to 59.	59
%n	The line feed.	Line feed
%p	The abbreviation that indicates the morning or afternoon. Valid values: AM and PM.	AM or PM
%r	The time. The 12-hour clock is used. The format is equivalent to %I:%M:%S %p.	11:59:59 AM
%R	The time. Hours and minutes are included. The format is equivalent to %H:%M.	23:59
%S	The second. The value is in the decimal format. Valid values: 00 to 59.	59
%t	The tab character.	None
%y	The two-digit number of the year. The value is in the decimal format. Valid values: 00 to 99.	04 or 98

Time format	Description	Example
%Y	The four-digit number of the year. The value is in the decimal format.	2004 or 1998
%C	The two-digit number of the century. The value is in the decimal format. Valid values: 00 to 99.	16
%e	The day of the month. The value is in the decimal format. Valid values: 1 to 31. Prefix a single-digit number with a space.	7 or 31
%j	The day of the year. The value is in the decimal format. Valid values: 001 to 366.	365
%u	The day of the week. The value is in the decimal format. Valid values: 1 to 7. The value 1 indicates Monday.	2
%U	The week of the year. Sunday is the first day of each week. Valid values: 00 to 53.	23
%V	The week of the year. Monday is the first day of each week. Valid values: 01 to 53. If a week on which January 1 falls has four or more days in January, the week is considered the first week of the year. Otherwise, the following week is considered the first week of the year.	24
%w	The day of the week. The value is in the decimal format. Valid values: 0 to 6. The value 0 indicates Sunday.	5
%W	The week of the year. Monday is the first day of each week. Valid values: 00 to 53.	23
%c	The date and time that follows the ISO 8601 standard.	Tue Nov 20 14:12:58 2020

Time format	Description	Example
%x	The date that follows the ISO 8601 standard.	Tue Nov 20 2020
%X	The time that follows the ISO 8601 standard.	11:59:59
%s	The UNIX timestamp.	1476187251

Examples

The following table lists commonly used time standards and time expressions, and provides related examples.

Example	Time expression	Time standard
2017-12-11 15:05:07	%Y-%m-%d %H:%M:%S	User-defined
[2017-12-11 15:05:07.012]	[%Y-%m-%d %H:%M:%S]	User-defined
02 Jan 06 15:04 MST	%d %b %y %H:%M	RFC822
02 Jan 06 15:04 -0700	%d %b %y %H:%M	RFC822Z
Monday, 02-Jan-06 15:04:05 MST	%A, %d-%b-%y %H:%M:%S	RFC850
Mon, 02 Jan 2006 15:04:05 MST	%A, %d %b %Y %H:%M:%S	RFC1123
2006-01-02T15:04:05Z07:00	%Y-%m-%dT%H:%M:%S	RFC3339
2006-01-02T15:04:05.999999999Z07:00	%Y-%m-%dT%H:%M:%S	RFC3339Nano
1637843406	%s	User-defined
1637843406123	%s	User-defined (Log Service considers second as the precision of the time.)

3.5.11. Log topics

Logs can be identified based on log topics. You can specify a topic generation mode when you create a Logtail configuration in the Log Service console. You can specify log topics when you upload logs by calling API operations or by using SDKs. This topic describes the topic generation modes that you can specify in the Log Service console.

- Null - Do not generate topic
- Machine Group Topic Attributes
- File Path RegEx
- Static topic generation

Null - Do not generate topic

If you set **Topic Generation Mode** to **Null - Do not generate topic**, no log topics are generated.

Machine Group Topic Attributes

This mode is used to distinguish the logs that are generated by different servers. If paths to the logs that are generated by different servers are the same or the files that store the logs are named the same, you can distinguish the logs based on log topics.

You can add servers to different machine groups and configure different topic attributes for the machine groups. Then, you can set **Topic Generation Mode** to **Machine Group Topic Attributes** when you create a Logtail configuration. When Logtail uploads the logs of servers in different machine groups to Log Service, Logtail includes the topic attributes of the machine groups as log topics. You can use log topics to query the logs.

File Path RegEx

This mode is used to distinguish the logs that are generated for different users or instances. If logs generated for different users or instances are stored in different directories but duplicate sub-directory names or log file names exist in these directories, Logtail cannot identify the user or instance for which the logs are generated when Logtail collects the logs.

Extract a value from a log file path

When you create a Logtail configuration, you can set **Topic Generation Mode** to **File Path RegEx**, specify a regular expression to match a log file path, and then use a capturing group to capture the content that you want to extract. You must specify a regular expression that can match the complete log file path. Only one capturing group is supported. When Logtail uploads the logs of different users or instances to Log Service, Logtail includes the names of the users or instances as log topics. You can use log topics to query the logs.

> **Note** You must escape the forward slash (/) in a regular expression that is used to match a log file path.

For example, the logs that are generated for different users are stored in different directories. However, duplicate log file names exist in the directories. Example:

```
/logs
   - /userA/serviceA
     - service.log
   - /userB/serviceA
     - service.log
   - /userC/serviceA
     - service.log
```

If you specify only */logs* for the log file path and specify *service.log* for the log file name in a Logtail configuration, Logtail collects logs from all files named *service.log* to a Logstore. The user for which the collected logs are generated cannot be identified. You can specify the following regular expression to extract a value from each log file path. Each value is used as a unique log topic.

- Regular expression

  ```
  \/(.*)\/serviceA\/.*
  ```

- Extraction results

```
__topic__: userA
__topic__: userB
__topic__: userC
```

Extract multiple values from a log file path

If the source of a log cannot be identified by a single log topic, you can configure multiple capturing groups in the regular expression that you specify. This way, Logtail can extract multiple values from a log file path and use the values as log tags to identify the source of the log. Capturing groups include named capturing groups and unnamed capturing groups. A named capturing group is in the ?P<name> format. If the capturing groups in the specified regular expression are all named capturing groups, tag fields are generated in the __tag__:{name} format. If the capturing groups in the specified regular expression are all unnamed capturing groups, tag fields are generated in the __tag__:__topic_{i}__ format. {i} indicates the ordinal number of an unnamed capturing group.

> **Note** If a specified regular expression includes multiple capturing groups, the __topic__ field is not generated.

For example, the log file path is */logs/userA/serviceA/service.log*. You can specify one of the following regular expressions to extract multiple values from the log file path.

- Example 1: Extract multiple values by using unnamed capturing groups in a regular expression.
  - Regular expression

    ```
    \/logs\/(.*?)\/(.*?)\/service.log
    ```

  - Extraction results

    ```
    __tag__:__topic_1__: userA
    __tag__:__topic_2__: serviceA
    ```

- Example 2: Extract multiple values by using named capturing groups in a regular expression.
  - Regular expression

    ```
    \/logs\/(?P<user>.*?)\/(?P<service>.*?)\/service.log
    ```

  - Extraction results

    ```
    __tag__:user: userA
    __tag__:service: serviceA
    ```

Static topic generation

You can set **Topic Generation Mode** to **File Path RegEx** and specify `customized:// + Custom topic name` in the **Custom RegEx** field. This way, custom static topics are used.

> **Note** This setting is supported on a Linux server on which Logtail V0.16.21 or later is installed.

3.6. Container log collection

3.6.1. Overview

Log Service allows you to collect Kubernetes container logs in either the DaemonSet or Sidecar mode. This topic describes the procedures and differences of log collection in the two modes.

Collection methods

Log collection in the DaemonSet mode features simple O&M, low resource usage, and flexible collection of container stdout files, stderr files, and text files. However, in the DaemonSet mode, Logtail collects logs from all containers on a node. This mode has performance bottlenecks and causes a loosely isolated environment for business logs. In the Sidecar mode, a Sidecar container is created for each container from which logs are to be collected. This improves multi-tenant isolation and performance.

Log collection configurations

You can use the Log Service console or use custom resource definitions (CRDs) to configure log collection. The following table lists the differences between CRDs and the console in terms of log collection configurations.

Item	CRD	Console
Procedure sophistication	Low	Moderate
Feature	Supports all configurations that the console supports and advanced configurations that the console does not support	Moderate
Ease of use	Moderate	Low
Network connection	Connects to a Kubernetes cluster	Connects to the Internet
Integration with container components	Supported	Unsupported
Authentication method	Authentication method provided by Kubernetes	Authentication method based on Alibaba Cloud accounts

Collection process

The following describes the log collection process in the DaemonSet method.

1. Install Logtail components.
2. Create collection configurations.

   Log Service allows you to configure log collection by using CRDs and the console to collect container logs of Kubernetes clusters.

   - Use CRDs to collect container logs in DaemonSet mode.
   - Use the Log Service console to collect container text logs in DaemonSet mode.
   - Use the Log Service console to collect container stdout and stderr in DaemonSet mode.

> **Note** If you use CRDs to configure log collection, projects, Logstores, indexes, server groups, Logtail configurations, and other resources are automatically created. In addition, log collection configurations better integrate with Kubernetes clusters. If you are configuring log collection from containers for the first time, you can use the console because the operation is easier.

The following describes the log collection process in the Sidecar method.

1. Install Logtail components.
2. Install Sidecars and create log collection configurations.

   Log service allows you to create collection configurations by using CRDs and the console to collect container logs of Kubernetes clusters.

   - Use CRDs to collect container text logs in Sidecar mode.
   - Use the Log Service console to collect container text logs in Sidecar mode.

3.6.2. Install Logtail components

This topic describes how to install Logtail components in a Kubernetes cluster.

Context

To collect container logs from a Kubernetes cluster, you must install Logtail components.

When you install Logtail components, the system automatically completes the following operations:

1. Create a ConfigMap named alibaba-log-configuration. The ConfigMap contains the configuration information of Log Service, such as projects.
2. Optional. Create a Custom Resource Definition (CRD) named AliyunLogConfig.
3. Optional. Create a Deployment controller named alibaba-log-controller. The Deployment controller is used to monitor the changes in the AliyunLogConfig CRD and the creation of Logtail configurations.
4. Create a DaemonSet named logtail-ds to collect logs from nodes.

Alibaba Cloud Container Service for Kubernetes (ACK) clusters

You can install Logtail components in an existing ACK cluster. You can also install Logtail components when you create an ACK cluster. To install Logtail components when you create an ACK cluster, you must select **Enable Log Service**.

Install Logtail components in an existing ACK cluster

> **Notice** If your ACK cluster is a dedicated or managed Kubernetes cluster, you can follow the instructions in this section to install Logtail components in your ACK cluster.

1.
2. In the left-side navigation pane, click **Clusters**.
3. On the **Clusters** page, find and click the cluster in which you want to install Logtail components.
4. In the left-side navigation pane of the page that appears, choose **Operations > Add-ons**.
5. On the **Logs and Monitoring** tab, find the **logtail-ds** component and click **Install**.

After the component is installed, a machine group named `k8s-group-${your_k8s_cluster_id}` and a Logstore named `config-operation-log` are automatically created in the project that you use.

> **Notice** Do not delete the `config-operation-log` Logstore.

Install Logtail components when you create an ACK cluster

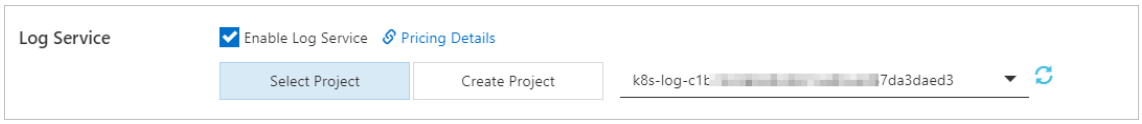
1.
2. In the left-side navigation pane, click **Clusters**.
3. On the **Clusters** page, click **Create Cluster**.
4. In the **Component Configurations** step, select **Enable Log Service**.

> **Note** This example shows how to enable Log Service. For information about how to create an ACK cluster, see Create an ACK managed cluster.

If you select **Enable Log Service**, the system prompts you to create a Log Service project. For information about how logs are managed in Log Service, see Project. You can use one of the following methods to create a project:

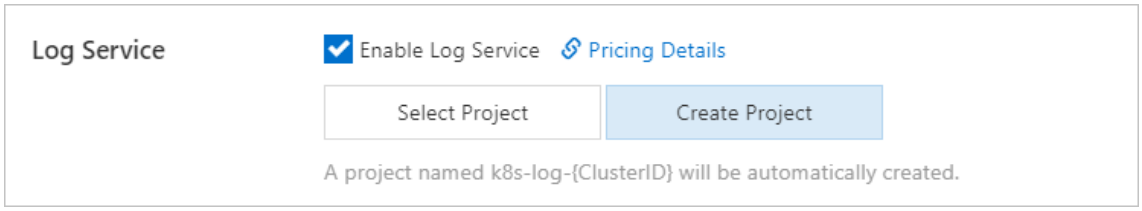
- **Select Project**

  You can select an existing project to manage the container logs that are collected.

- **Create Project**

  Log Service automatically creates a project named `k8s-log-{ClusterID}` to manage the container logs that are collected. `ClusterID` indicates the unique ID of the ACK cluster that you create.

After the component is installed, a machine group named `k8s-group-${your_k8s_cluster_id}` and a Logstore named `config-operation-log` are automatically created in the project that you use.

> **Notice** Do not delete the `config-operation-log` Logstore.

Self-managed Kubernetes clusters

1.
2. Create a project whose name starts with `k8s-log-custom-`.

   Example: k8s-log-custom-sd89ehdq. For more information, see Create a project.

3. Log on to your Kubernetes cluster.
4. Run the following commands to install Logtail and dependent components.

> **Notice**
>
> - Make sure that the kubectl command-line tool is installed on the machine on which you want to run the commands.
> - The alibaba-log-controller Deployment controller is available only in Kubernetes 1.6 or later.
> - If you no longer need to use CRDs, you can delete the *alibaba-cloud-log/templates/alicloud-log-config.yaml* file and rerun the following commands. If the `./alicloud-log-k8s-custom-install.sh: line 111: /root/alibaba-cloud-log/templates/alicloud-log-crd.yaml: No such file or directory` error message appears, you can ignore the error.

i. Download the installation script.

```
wget http://logtail-release-cn-hangzhou.oss-cn-hangzhou.aliyuncs.com/kubernetes/alicloud-log-k8s-custom-install.sh
```

ii. Modify permissions to limit access to the installation script.

```
chmod 744 ./alicloud-log-k8s-custom-install.sh
```

iii. Install Logtail and dependent components.

```
sh ./alicloud-log-k8s-custom-install.sh your-project-suffix region-id aliuid access
-key-id access-key-secret
```

The following table describes the parameters that are included in the preceding command. You can configure the parameters based on your business scenario.

Parameter	Description
your-project-suffix	The part that follows k8s-log-custom- in the name of your project. Use the project that you created in Step . For example, if the project name is `k8s-log-custom-sd89ehdq`, set this parameter to `sd89ehdq`.
region-id	The ID of the region where your project resides. For example, the ID of the China (Hangzhou) region is `cn-hangzhou`. For more information, see Supported regions.
aliuid	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.
access-key-id	The AccessKey ID of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user and attach the AliyunLogFullAccess policy to the RAM user. For more information, see Create a RAM user and authorize the RAM user to access Log Service.
access-key-secret	The AccessKey secret of your Alibaba Cloud account. We recommend that you use the AccessKey pair of a RAM user and attach the AliyunLogFullAccess policy to the RAM user. For more information, see Create a RAM user and authorize the RAM user to access Log Service.

After the components are installed, a machine group named `k8s-group-${your_k8s_cluster_id}` and a Logstore named `config-operation-log` are automatically created in the project that you use.

Notice

- Do not delete the `config-operation-log` Logstore.
- If you install the components in a self-managed Kubernetes cluster, Logtail is granted the `privileged` permissions by default. This prevents the `container text file busy` error that may occur when other pods are deleted. For more information, see Bug 1468249, Bug 1441737, and Issue 34538.

FAQ

- How do I view the version of a container image?

  You can view the version of a container image by visiting the following image repository: https://cr.console.aliyun.com/repository/cn-shanghai/log-service/logtail/images.

- How do I collect container logs from multiple Kubernetes clusters to the same Log Service project?
  - Alibaba Cloud ACK clusters

    If you want to collect container logs from multiple ACK clusters to the same Log Service project, you must select the same project when you create the ACK clusters.
  - Self-managed Kubernetes clusters

    If you want to collect container logs from multiple self-managed Kubernetes clusters to the same Log Service project, you must set the {your-project-suffix} parameter to the same value that you specified when you install Logtail components in each Kubernetes cluster.

  > **Note** You can collect container logs from multiple self-managed Kubernetes clusters to the same Log Service project only if the Kubernetes clusters reside in the same region.

- How do I view the logs of Logtail?

  The logs of Logtail are stored in the *ilogtail.LOG* and *logtail_plugin.LOG* files in the */usr/local/ilogtail/* directory of a Logtail container.

  The standard output of the container is irrelevant to this case. Ignore the following standard output:

```
start umount useless mount points, /shm$|/merged$|/mqueue$
umount: /logtail_host/var/lib/docker/overlay2/3fd0043af174cb0273c3c7869500fbe2bdb95d13b1e
110172ef57fe840c82155/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/d5b10aa19399992755de1f85d25009528daa749c1bf
8c16edff44beab6e69718/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/5c3125daddacedec29df72ad0c52fac800cd56c6e88
0dc4e8a640b1e16c22dbe/merged: must be superuser to unmount
......
xargs: umount: exited with status 255; aborting
umount done
start logtail
ilogtail is running
logtail status:
ilogtail is running
```

- How do I view the status of Log Service components in Kubernetes clusters?

  Run the following commands:

```
kubectl get deploy alibaba-log-controller -n kube-system
kubectl get ds logtail-ds -n kube-system
```

- What do I do if alibaba-log-controller fails to start?

  Check whether alibaba-log-controller is installed by using the following method:

  - Run the installation command on the master node of your Kubernetes cluster.
  - Specify the ID of your Kubernetes cluster in the installation command.

  If alibaba-log-controller is not installed by using the preceding method, run the `kubectl delete -f deploy` command to delete the installation template that is generated. Then, rerun the installation command.

- How do I view the status of the Logtail DaemonSet in a Kubernetes cluster?

Run the `kubectl get ds -n kube-system` command to view the status of the Logtail DaemonSet.

> **Note** The default namespace in which a Logtail container resides is kube-system.

- How do I view the version number, IP address, startup time, and status of Logtail?
  - Run the following command to view the status of Logtail:

    ```
    kubectl get po -n kube-system | grep logtail
    ```

    The following output is returned:

    ```
    NAME            READY     STATUS    RESTARTS   AGE
    logtail-ds-gb92k   1/1       Running   0          2h
    logtail-ds-wm7lw   1/1       Running   0          4d
    ```

  - Run the following command to view the version number and IP address of Logtail:

    ```
    kubectl exec logtail-ds-gb92k -n kube-system cat /usr/local/ilogtail/app_info.json
    ```

    The following output is returned:

    ```
    {
       "UUID" : "",
       "hostname" : "logtail-ds-gb92k",
       "instance_id" : "0EBB2B0E-0A3B-11E8-B0CE-0A58AC140402_172.20.4.2_1517810940",
       "ip" : "192.0.2.0",
       "logtail_version" : "0.16.2",
       "os" : "Linux; 3.10.0-693.2.2.el7.x86_64; #1 SMP Tue Sep 12 22:26:13 UTC 2017; x86_6
    4",
       "update_time" : "2021-02-05 06:09:01"
    }
    ```

- How do I view the operational logs of Logtail?

  The operational logs of Logtail are stored in the *ilogtail.LOG* file in the */usr/local/ilogtail/* directory. If the log file is rotated, the generated files are compressed and stored as *ilogtail.LOG.x.gz*.

  Example:

  ```
  [root@iZbp1dsu6v77zfb40qfbiaZ ~]# kubectl exec logtail-ds-gb92k -n kube-system tail /usr/
  local/ilogtail/ilogtail.LOG
  [2018-02-05 06:09:02.168693] [INFO] [9] [build/release64/sls/ilogtail/LogtailPlugin.cpp:1
  04] logtail plugin Resume:start
  [2018-02-05 06:09:02.168807] [INFO] [9] [build/release64/sls/ilogtail/LogtailPlugin.cpp:1
  06] logtail plugin Resume:success
  [2018-02-05 06:09:02.168822] [INFO] [9] [build/release64/sls/ilogtail/EventDispatcher.cpp
  :369] start add existed check point events, size:0
  [2018-02-05 06:09:02.168827] [INFO] [9] [build/release64/sls/ilogtail/EventDispatcher.cpp
  :511] add existed check point events, size:0 cache size:0 event size:0 success count:0
  ```

- How do I restart Logtail for a pod?

  i. Stop Logtail.

  In the following command, `logtail-ds-gb92k -n` specifies the container, and `kube-system` specifies the namespace. Configure the parameters based on your business scenario.

```
kubectl exec logtail-ds-gb92k -n kube-system /etc/init.d/ilogtaild stop
```

If the following output is returned, Logtail is stopped.

```
kill process Name: ilogtail pid: 7
kill process Name: ilogtail pid: 9
stop success
```

ii. Start Logtail.

In the following command, `logtail-ds-gb92k -n` specifies the container, and `kube-system` specifies the namespace. Configure the parameters based on your business scenario.

```
kubectl exec logtail-ds-gb92k -n kube-system /etc/init.d/ilogtaild start
```

If the following output is returned, Logtail is started.

```
ilogtail is running
```

What's next

Create Logtail configurations to collect container logs.

- DaemonSet mode
  - For information about how to collect container logs by using CRDs, see Use CRDs to collect container logs in DaemonSet mode.
  - For information about how to collect container stdout and stderr logs by using the Log Service console, see Use the Log Service console to collect container stdout and stderr in DaemonSet mode.
  - For information about how to collect container text logs by using the Log Service console, see Collect log data from containers by using Log Service.
- Sidecar mode
  - For information about how to collect container logs by using CRDs, see Use CRDs to collect container text logs in Sidecar mode.
  - For information about how to collect container logs by using the Log Service console, see Use the Log Service console to collect container text logs in Sidecar mode.

3.6.3. Use the Log Service console to collect container text logs in DaemonSet mode

This topic describes how to create a Logtail configuration in the Log Service console and use the Logtail configuration to collect container text logs in DaemonSet mode.

Prerequisites

The Logtail component is installed. For more information, see Install Logtail components.

Features

Logtail can collect container text logs, and then upload the text logs together with container metadata to Log Service. Logtail supports the following features:

- Allows you to specify a log file path in a container. You do not need to manually map the log file path to a path on the host.
- Uses the container label whitelist to specify containers from which text logs are collected.
- Uses the container label blacklist to specify containers from which text logs are not collected.
- Uses the environment variable whitelist to specify containers from which text logs are collected.
- Uses the environment variable blacklist to specify containers from which text logs are not collected.
- Collects multi-line logs. For example, Logtail can collect Java stack logs.
- Automatically associates container metadata that needs to be uploaded together with the collected container text logs. The metadata includes container names, image names, pod names, namespaces, and environment variables.
- If a container runs in a Kubernetes cluster, Logtail also supports the following features:
  - Uses Kubernetes namespaces, pod names, and container names to specify containers from which text logs are collected.
  - Uses the Kubernetes label whitelist to specify containers from which text logs are collected.
  - Uses the Kubernetes label blacklist to specify containers from which text logs are not collected.
  - Automatically associates Kubernetes labels that need to be uploaded together with the collected container text logs.

Limits

- If Logtail detects the `die` event on a container that is stopped, Logtail no longer collects text logs from the container. If collection latency exists, some text logs that are collected before the container is stopped may be lost.
- For Docker containers, only overlay and overlay2 storage drivers are supported. If other storage drivers are used, you must mount a volume to the directory of logs. Then, a temporary directory is generated.

  If an Apsara File Storage NAS (NAS) file system is mounted to the directory of logs by using a PersistentVolumeClaim (PVC), you cannot collect logs in DaemonSet mode. In this case, we recommend that you collect logs in Sidecar mode.
- Logtail cannot access the symbolic link of a container. You must specify an actual path as the collection directory.
- If a volume is mounted to the data directory of a container, Logtail cannot collect data from the parent directory of the data directory. You must specify the complete path of the data directory as the collection directory.

  For example, if a volume is mounted to the */var/log/service* directory and you set the collection directory to */var/log*, Logtail cannot collect logs from the /var/log directory. You must specify */var/log/service* as the collection directory.
- By default, Kubernetes mounts the root directory of the host to the `/logtail_host` directory of the Logtail container. If you want to collect text logs from the host, you must specify `/logtail_host` as the prefix of the log file path.

  For example, if you want to collect logs from the `/home/logs/app_log/` directory of the host, you must specify `/logtail_host/home/logs/app_log/` as the log file path.
- Logtail collects data from containers that use the Docker engine or containerd engine.
  - Docker: Logtail accesses the Docker engine in the */run/docker.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.

- containerd: Logtail accesses the containerd engine in the */run/containerd/containerd.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.

Create a Logtail configuration

1.
2. In the **Import Data** section, click **Kubernetes - Object**.
3. Select a project and a Logstore. Then, click **Next**.

   In this example, select the project that you use to install the Logtail component and the Logstore that you create.
4. Click **Use Existing Machine Groups**.

   After you install the Logtail component, Log Service automatically creates a machine group named `k8s-group-${your_k8s_cluster_id}`. You can select this machine group.
5. Select the `k8s-group-${your_k8s_cluster_id}` machine group from **Source Server Groups** and move the machine group to **Applied Server Groups**. Then, click **Next**.

   > **Notice** If the heartbeat status of the machine group is **FAIL**, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
6. Configure the parameters for the Logtail configuration and click **Next**.
   i. Configure the basic settings, such as the name, log path, and mode. For more information, see Collect text logs.
   ii. Turn on **Docker File**.
   iii. (Optional)Specify conditions to filter containers.
   - For versions earlier than Logtail V1.0.29, containers can be filtered only by using environment variables and container labels.

     A namespace of a Kubernetes cluster and the name of a container in a Kubernetes cluster can be mapped to container labels. The value of the LabelKey parameter for a namespace is `io.kubernetes.pod.namespace`. The value of the LabelKey parameter for a container name is `io.kubernetes.container.name`. We recommend that you use the two container labels to filter containers. If the container labels do not meet your business requirements, you can use the environment variable whitelist or the environment variable blacklist to filter containers. For example, the namespace of a pod is backend-prod, and the name of a container in the pod is worker-server. If you want the logs of the worker-server container to be collected, you can specify `io.kubernetes.pod.namespace : backend-prod` or `io.kubernetes.container.name : worker-server` in the container label whitelist.

Notice

- Container labels are retrieved by running the docker inspect command. Container labels are different from Kubernetes labels. For more information, see Obtain container labels.
- Environment variables are the same as the environment variables that are configured to start containers. For more information, see Obtain environment variables.
- Do not specify duplicate values for the LabelKey parameter. If you specify duplicate values for the LabelKey parameter, only one of the values takes effect.

Parameter	Description
Label Whitelist	The container label whitelist. The whitelist specifies the containers from which text logs are collected. When you configure the container label whitelist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose container labels contain the keys specified by LabelKey are matched. ■ If you specify a value for the LabelValue parameter, containers whose container labels consist of the key-value pair specified by LabelKey and LabelValue are matched. By default, string matching is performed for the values of the LabelValue parameter. Containers are matched only if the values of the container labels are the same as the values of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the LabelKey parameter to *io.kubernetes.container.name* and set the LabelValue parameter to *^(nginx\|cube)$*, a container named nginx and a container named cube are matched. Key-value pairs are connected by using the OR operator. If a container label consists of one of the specified key-value pairs, the container to which the container label belongs is matched.

Parameter	Description
Label Blacklist	The container label blacklist. The blacklist specifies the containers from which text logs are not collected. When you configure the container label blacklist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose container labels contain the keys specified by LabelKey are filtered out. ■ If you specify a value for the LabelValue parameter, containers whose container labels consist of the key-value pair specified by LabelKey and LabelValue are filtered out. By default, string matching is performed for the values of the LabelValue parameter. Containers are matched only if the values of the container labels are the same as the values of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the LabelKey parameter to *io.kubernetes.container.name* and set the LabelValue parameter to *^(nginx\|cube)$*, a container named nginx and a container named cube are matched. Key-value pairs are connected by using the OR operator. If a container label consists of one of the specified key-value pairs, the container to which the container label belongs is filtered out.

Parameter	Description
Environment Variable Whitelist	The environment variable whitelist. The whitelist specifies the containers from which text logs are collected. When you configure the environment variable whitelist, the EnvKey parameter is required, and the EnvValue parameter is optional. ■ If you leave the EnvValue parameter empty, containers whose environment variables contain the keys specified by EnvKey are matched. ■ If you specify a value for the EnvValue parameter, containers whose environment variables consist of the key-value pair specified by EnvKey and EnvValue are matched. By default, string matching is performed for the values of the EnvValue parameter. Containers are matched only if the values of the environment variables are the same as the values of the EnvValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the EnvValue parameter, regular expression matching is performed. For example, if you set the EnvKey parameter to *NGINX_SERVICE_PORT* and set the EnvValue parameter to *^(80\|6379)$*, containers whose port number is 80 and containers whose port number is 6379 are matched. Key-value pairs are connected by using the OR operator. If an environment variable consists of one of the specified key-value pairs, the container to which the environment variable belongs is matched.

Parameter	Description
Environment Variable Blacklist	The environment variable blacklist. The blacklist specifies the containers from which text logs are not collected. When you configure the environment variable blacklist, the EnvKey parameter is required, and the EnvValue parameter is optional. ■ If you leave the EnvValue parameter empty, containers whose environment variables contain the keys specified by EnvKey are filtered out. ■ If you specify a value for the EnvValue parameter, containers whose environment variables consist of the key-value pair specified by EnvKey and EnvValue are filtered out. By default, string matching is performed for the values of the EnvValue parameter. Containers are matched only if the values of the environment variables are the same as the values of the EnvValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the EnvValue parameter, regular expression matching is performed. For example, if you set the EnvKey parameter to *NGINX_SERVICE_PORT* and set the EnvValue parameter to *^(80\|6379)$*, containers whose port number is 80 and containers whose port number is 6379 are matched. Key-value pairs are connected by using the OR operator. If an environment variable consists of one of the specified key-value pairs, the container to which the environment variable belongs is filtered out.

■ For Logtail V1.0.29 or later, we recommend that you use different levels of Kubernetes information, such as pod names, namespaces, container names, and labels to filter containers.

Turn on **Deployed in K8s** and configure the following parameters to filter containers.

> **Note** If you change Kubernetes labels when Kubernetes control resources, such as Deployments, are running, the operational pod is not restarted. Therefore, the pod cannot detect the change. This may cause a matching rule to become invalid. When you specify the Kubernetes label whitelist and the Kubernetes label blacklist, we recommend that you use the Kubernetes labels of pods.

Parameter	Description

Parameter	Description
K8s Pod Name Regular Matching	The pod name. The pod name specifies the containers from which text logs are collected. Regular expression matching is supported. For example, if you specify *^(nginx-log-demo.*)$*, all containers in the pod whose name starts with nginx-log-demo are matched.
K8s Namespace Regular Matching	The namespace. The namespace specifies the containers from which text logs are collected. Regular expression matching is supported. For example, if you specify *^(default|nginx)$*, all containers in the nginx and default namespaces are matched.
K8s Container Name Regular Matching	The container name. The container name specifies the containers from which text logs are collected. Regular expression matching is supported. Kubernetes container names are defined in spec.containers. For example, if you specify *^(container-test)$*, all containers whose name is container-test are matched.
K8s Label Whitelist	The Kubernetes label whitelist. The whitelist specifies the containers from which text logs are collected. When you configure the Kubernetes label whitelist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose Kubernetes labels contain the keys specified by LabelKey are matched. ■ If you specify a value for the LabelValue parameter, containers whose Kubernetes labels consist of the key-value pair specified by LabelKey and LabelValue are matched. By default, string matching is performed for the values of the LabelValue parameter. Containers are matched only if the values of the Kubernetes labels are the same as the values of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to *^(test1\|test2)$*, containers whose Kubernetes labels consist of app:test1 or app:test2 are matched. Key-value pairs are connected by using the OR operator. If a Kubernetes label consists of one of the specified key-value pairs, the container to which the Kubernetes label belongs is matched.

Parameter	Description
K8s Label Blacklist	The Kubernetes label blacklist. The blacklist specifies the containers from which text logs are not collected. When you configure the Kubernetes label blacklist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose Kubernetes labels contain the keys specified by LabelKey are filtered out. ■ If you specify a value for the LabelValue parameter, containers whose Kubernetes labels consist of the key-value pair specified by LabelKey and LabelValue are filtered out. By default, string matching is performed for the values of the LabelValue parameter. Containers are matched only if the values of the Kubernetes labels are the same as the values of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to *^(test1\|test2)$*, containers whose Kubernetes labels consist of app:test1 or app:test2 are matched. Key-value pairs are connected by using the OR operator. If a Kubernetes label consists of one of the specified key-value pairs, the container to which the Kubernetes label belongs is filtered out.

iv. (Optional)Specify log labels.

For Logtail V1.0.29 or later, we recommend that you specify environment variables and Kubernetes labels for logs as log labels.

Parameter	Description
Environment Variable Log Tag	After you specify environment variables as log labels, Log Service adds environment variable-related fields to logs. For example, if you set the **EnvKey** parameter to *VERSION* and set the **EnvValue** parameter to *env_version*, Log Service adds the __tag__:__env_version__: v1.0.0 field to logs if the environment variable configurations of a container include `VERSION=v1.0.0` .
K8s Label Log Tag	After you specify Kubernetes labels as log labels, Log Service adds Kubernetes label-related fields to logs. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to `k8s_label_app` , Log Service adds the __tag__:__k8s_label_app__: serviceA field to logs if the label configurations of a Kubernetes cluster include `app=serviceA` .

7. Preview data, configure indexes, and then click **Next**.

By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

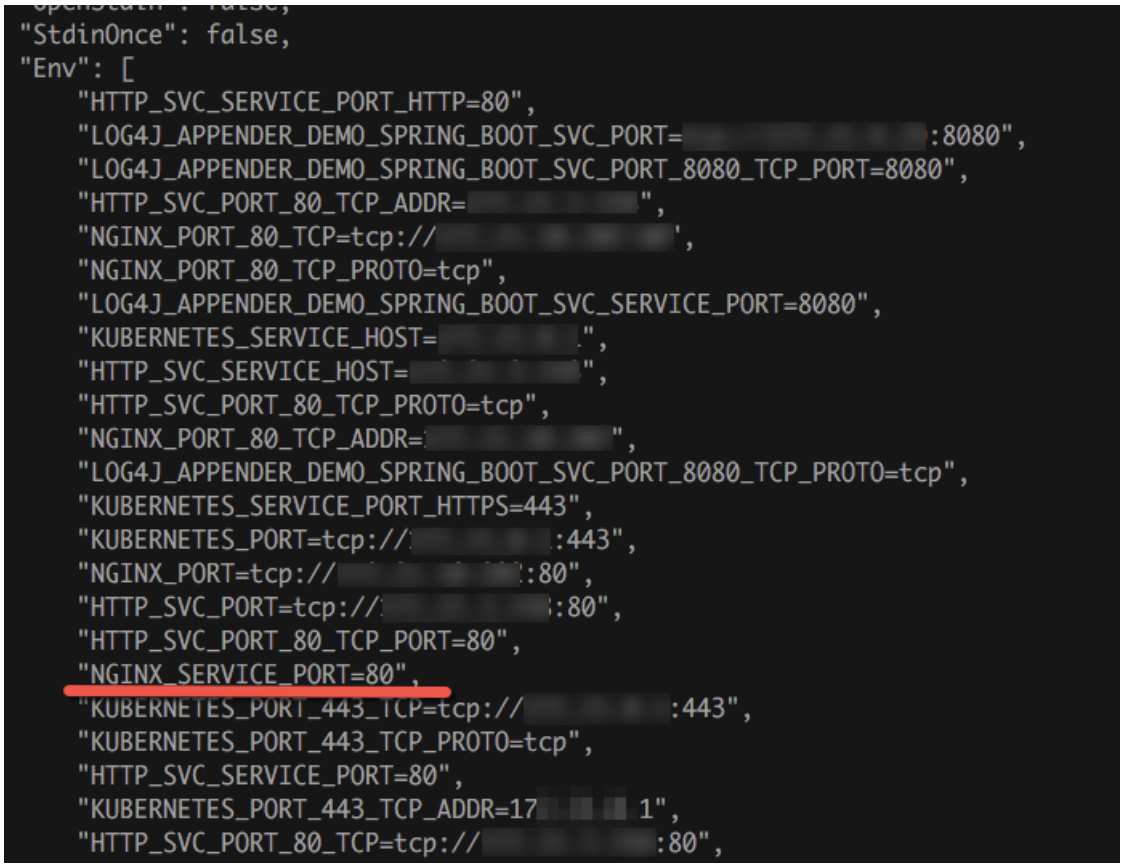
Configuration examples

Example 1: Filter containers based on the environment variable whitelist and the environment variable blacklist

Collect text logs from the containers whose environment variable configurations include `NGINX_SERVICE_PORT=80` but exclude `POD_NAMESPACE=kube-system`. The log file path is `/var/log/nginx/access.log`. The logs are parsed in **simple mode**.

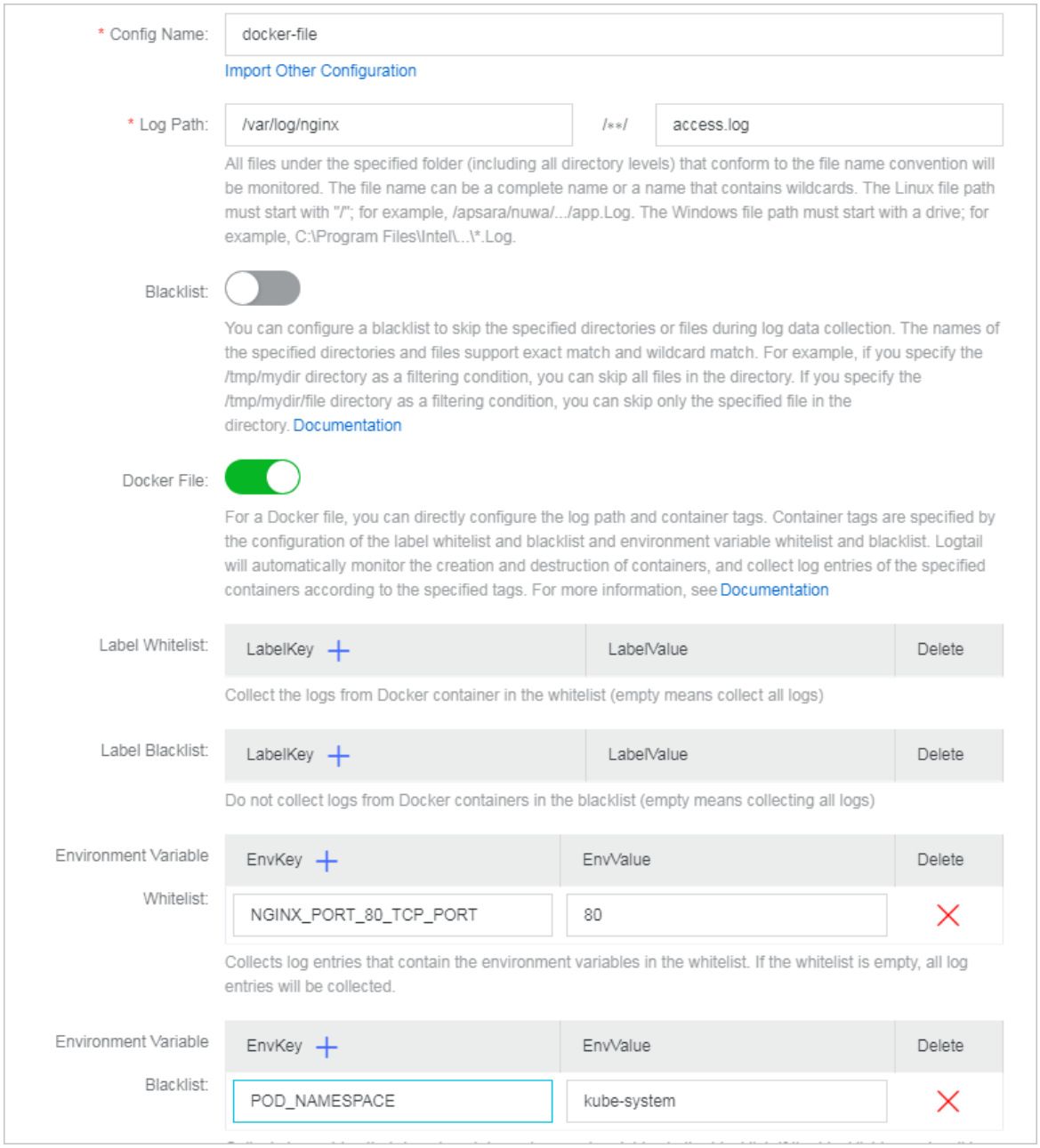
1. Obtain environment variables.

   To view the environment variables of a container, you can log on to the host on which the container resides. For more information, see Obtain environment variables.



2. Create a Logtail configuration.

The following figure shows an example of a Logtail configuration. For more information about how to create a Logtail configuration that is used to collect logs in simple mode, see Collect logs in simple mode.



Example 2: Filter containers based on the container label whitelist and the container label blacklist

Collect text logs from the containers whose container label is `io.kubernetes.container.name=nginx`. The log file path is `/var/log/nginx/access.log`. The logs are parsed in **simple mode**.

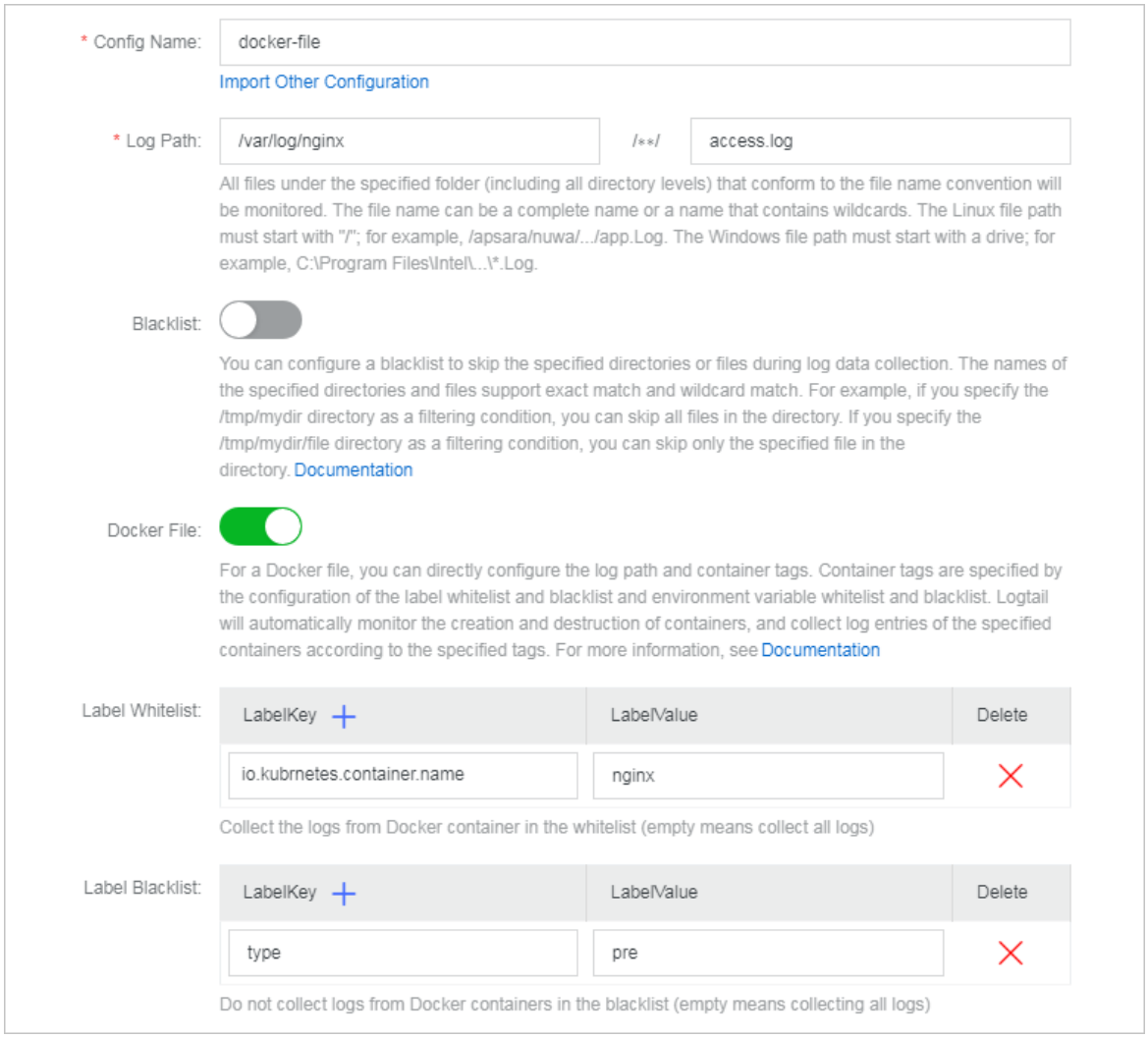
1. Obtain container labels.

   To view the container labels of a container, you can log on to the host on which the container resides. For more information, see Obtain container labels.

```
"OnBuild": null,
"Labels": {
    "annotation.io.kubernetes.container.hash": "53073f5a",
    "annotation.io.kubernetes.container.restartCount": "0",
    "annotation.io.kubernetes.container.terminationMessagePath": "/dev/termination-log",
    "annotation.io.kubernetes.container.terminationMessagePolicy": "File",
    "annotation.io.kubernetes.pod.terminationGracePeriod": "30",
    "io.kubernetes.container.logpath": "/var/log/pods/ad00a078-4182 585/nginx_0.log",
    "io.kubernetes.container.name": "nginx",
    "io.kubernetes.docker.type": "container",
    "io.kubernetes.pod.name": "example-foo-86ccd54874-r4mfh",
    "io.kubernetes.pod.namespace": "default",
    "io.kubernetes.pod.uid": "ad0
    "io.kubernetes.sandbox.id": "1dfa6da112969",
    "maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"
},
"StopSignal": "SIGTERM"
```


2. Create a Logtail configuration.

   The following figure shows an example of a Logtail configuration. For more information about how to create a Logtail configuration that is used to collect logs in simple mode, see Collect logs in simple mode.



Example 3: Filter containers by using Kubernetes namespaces, pod names, and container names

Collect text logs from the nginx-log-demo-0 container in pods whose name starts with nginx-log-demo in the default namespace.

1. Obtain different levels of Kubernetes information.
   - Obtain information about pods.

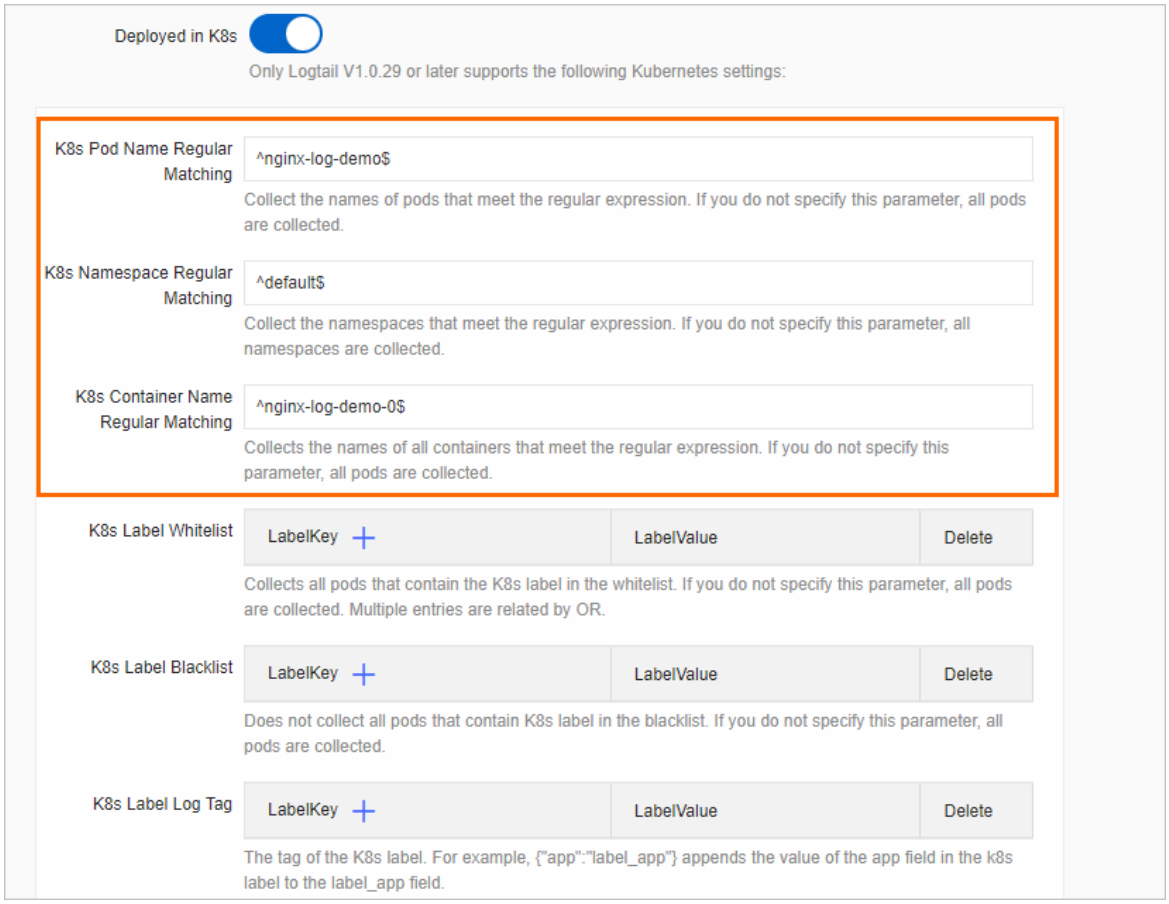
```
~/.kube » kubectl get pods
NAME                    READY   STATUS    RESTARTS   AGE
nginx-log-demo-0-bxl79  1/1     Running   0          48d
nginx-log-demo-1-qmrqk  1/1     Running   0          48d
nginx-log-demo-2-7khv9  1/1     Running   0          48d
nginx-log-demo-3-j24xc  1/1     Running   0          48d
```

   - Obtain information about namespaces.

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubernetes.io/psp: ack.privileged
  creationTimestamp: "2022-01-06T18:42:43Z"
  generateName: nginx-log-demo-0-
  labels:
    controller-uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
    job-name: nginx-log-demo-0
  name: nginx-log-demo-0-bxl79
  namespace: default
  ownerReferences:
  - apiVersion: batch/v1
    blockOwnerDeletion: true
    controller: true
    kind: Job
    name: nginx-log-demo-0
    uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
  resourceVersion: "50566856"
  uid: ee10fb7d-d989-47b3-bc2a-e9ffbe767849
spec:
  containers:
  - args:
    - --log-type=nginx
    - --stdout=true
    - --total-count=100000000000
    - --log-file-size=1000000000
    - --log-file-count=2
    - --logs-per-sec=2778
    command:
    - /bin/mock_log
    image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
    imagePullPolicy: Always
    name: nginx-log-demo-0
    resources:
```

2. Create a Logtail configuration.

   The following figure shows an example of a Logtail configuration. For more information about how to create a Logtail configuration that is used to collect logs in simple mode, see Collect logs in simple mode.



Example 4: Filter containers by using Kubernetes labels

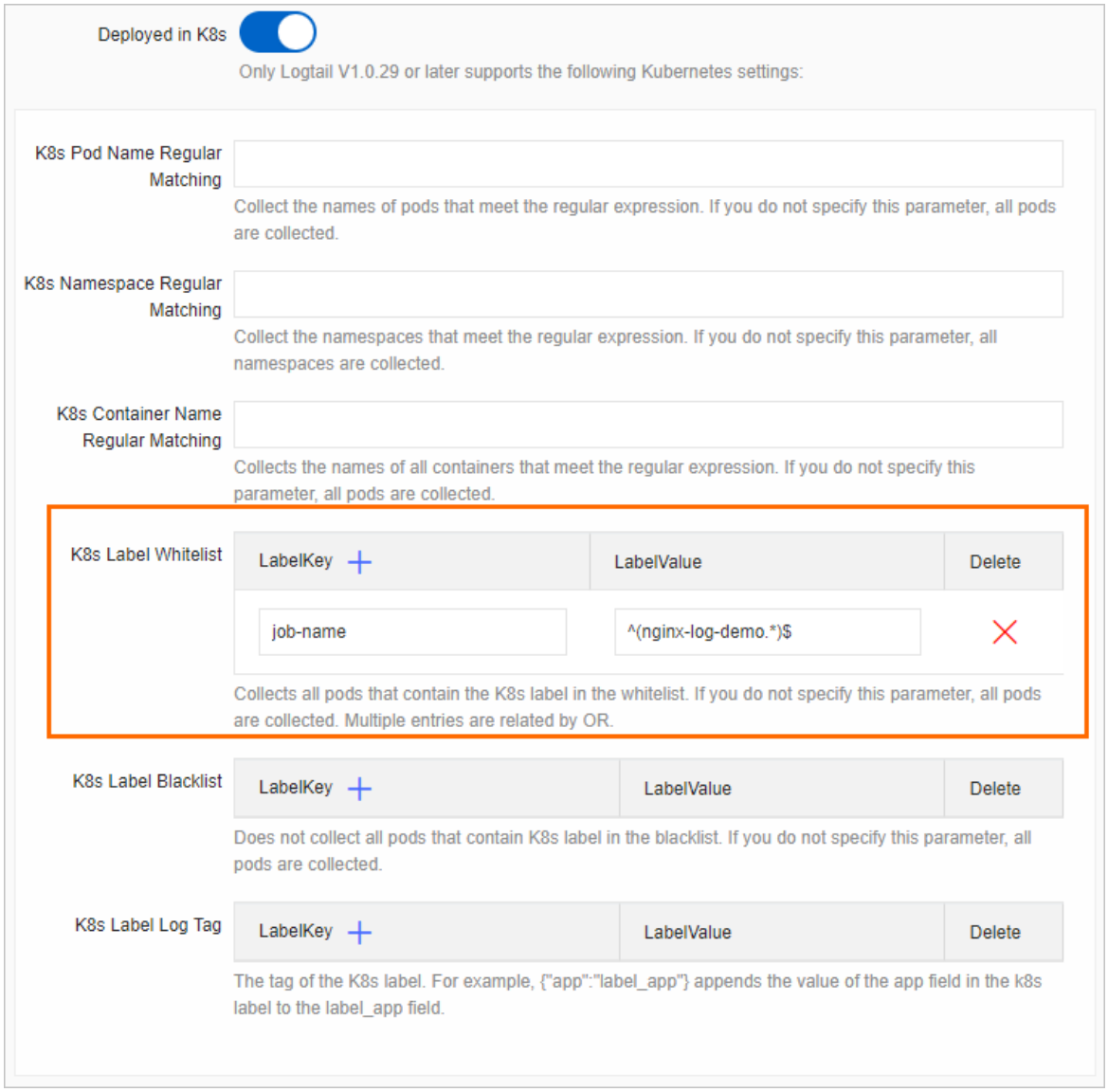
Collect text logs from containers whose Kubernetes labels contain the job-name key and a specific value. The value starts with nginx-log-demo.

1. Obtain Kubernetes labels.

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubernetes.io/psp: ack.privileged
  creationTimestamp: "2022-01-06T18:42:43Z"
  generateName: nginx-log-demo-0-
  labels:
    controller-uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
    job-name: nginx-log-demo-0
  name: nginx-log-demo-0-bx179
  namespace: default
  ownerReferences:
  - apiVersion: batch/v1
    blockOwnerDeletion: true
    controller: true
    kind: Job
    name: nginx-log-demo-0
    uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
  resourceVersion: "50566856"
  uid: ee10fb7d-d989-47b3-bc2a-e9ffbe767849
```

2. Create a Logtail configuration.

   The following figure shows an example of a Logtail configuration. For more information about how to create a Logtail configuration that is used to collect logs in simple mode, see Collect logs in simple mode.



Default fields

The following table describes the fields that are included by default in each container text log.

Log field	Description
_image_name_	The name of the image.
_container_name_	The name of the container.
_pod_name_	The name of the pod.
namespace	The namespace of the pod.

Log field	Description
_pod_uid_	The unique identifier of the pod.
_container_ip_	The IP address of the pod.

3.6.4. Use the Log Service console to collect container stdout and stderr in DaemonSet mode

This topic describes how to create a Logtail configuration in the Log Service console and use the Logtail configuration to collect container stdout and stderr in DaemonSet mode.

Prerequisites

- The Logtail component is installed. For more information, see Install Logtail components.
- A Logstore is created in the project that you use to install the Logtail component. For more information, see Create a Logstore.

Features

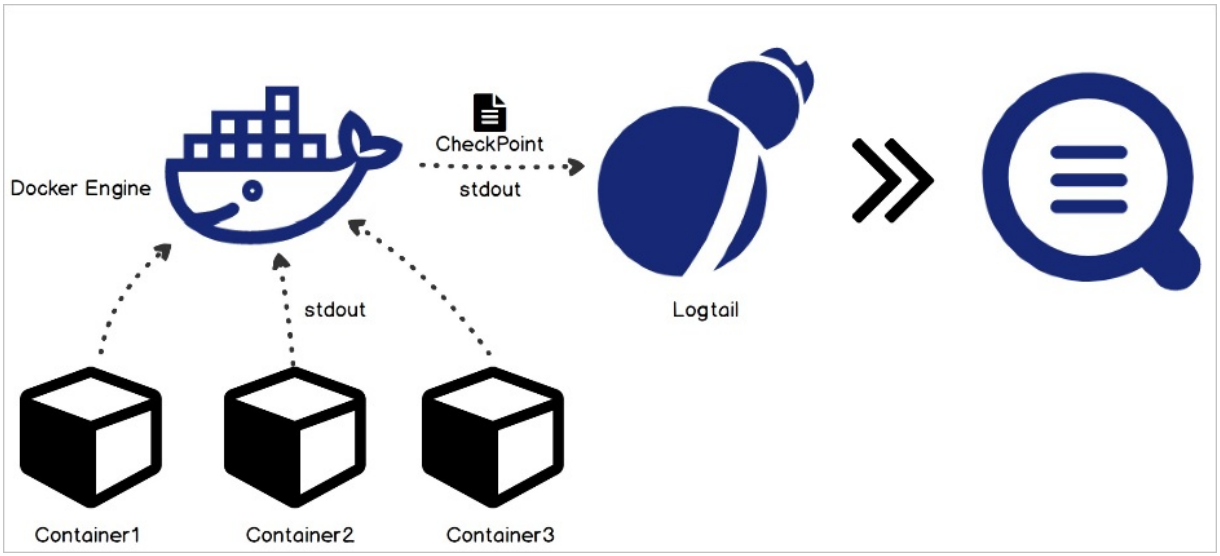
Logtail can collect container stdout and stderr, and then upload the stdout and stderr together with container metadata to Log Service. Logtail supports the following features:

- Collects stdout and stderr.
- Uses the container label whitelist to specify containers from which stdout and stderr are collected.
- Uses the container label blacklist to specify containers from which stdout and stderr are not collected.
- Uses the environment variable whitelist to specify containers from which stdout and stderr are collected.
- Uses the environment variable blacklist to specify containers from which stdout and stderr are not collected.
- Collects multi-line logs. For example, Logtail can collect Java stack logs.
- Automatically associates container metadata that needs to be uploaded together with the collected container stdout and stderr. The metadata includes container names, image names, pod names, namespaces, and environment variables.
- If a container runs in a Kubernetes cluster, Logtail also supports the following features:
  - Uses Kubernetes namespaces, pod names, and container names to specify containers from which stdout and stderr are collected.
  - Uses the Kubernetes label whitelist to specify containers from which stdout and stderr are collected.
  - Uses the Kubernetes label blacklist to specify containers from which stdout and stderr are not collected.
  - Automatically associates Kubernetes labels that need to be uploaded together with the collected container stdout and stderr.

Implementation

Logtail communicates with the domain socket of Docker. Logtail queries all Docker containers and identifies the containers from which stdout and stderr are collected by using the specified labels and environment variables. Logtail runs the `docker logs` command to collect logs from the specified containers.

When Logtail collects stdout and stderr from a container, Logtail periodically stores checkpoints to a checkpoint file. If Logtail is stopped and is then restarted, Logtail collects logs from the last checkpoint.



Limits

- You can use the Log Service console to collect stdout and stderr in DaemonSet mode only if Logtail runs V0.16.0 or later and runs on Linux. For more information about Logtail versions and version updates, see Install Logtail on a Linux server.
- Logtail collects data from containers that use the Docker engine or containerd engine.
  - Docker: Logtail accesses the Docker engine in the */run/docker.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.
  - containerd: Logtail accesses the containerd engine in the */run/containerd/containerd.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.
- By default, the last multi-line log that is collected by Logtail is cached for 3 seconds. This prevents the multi-line log from being split into multiple logs due to output latency. You can change the cache time by changing the value of the `BeginLineTimeoutMs` parameter. We recommend that you do not specify a value less than 1000 with millisecond precision. If you specify a value that is less than 1000, an error may occur.
- If Logtail detects the `die` event on a container that is stopped, Logtail no longer collects stdout and stderr from the container. If collection latency occurs, some stdout and stderr that are collected before the container is stopped may be lost.
- The logging driver collects stdout and stderr only in the JSON format from containers that use the Docker engine.
- By default, stdout and stderr that are collected from different containers by using the same Logtail configuration have the same context. If you want to specify a different context for the stdout and stderr of each container, you must create a Logtail configuration for each container.
- By default, the collected data is stored in the `content` field. Logtail can process the collected

data. For more information, see Use Logtail plug-ins to process data.

Create a Logtail configuration

1.
2. In the **Import Data** section, click **Kubernetes - Standard Output**.
3. Select a project and a Logstore. Then, click **Next**.

   Select the project that you use to install the Logtail component and the Logstore that you create.
4. Click **Use Existing Machine Groups**.

   After you install the Logtail component, Log Service automatically creates a machine group named `k8s-group-${your_k8s_cluster_id}`. You can select this machine group.
5. Select the `k8s-group-${your_k8s_cluster_id}` machine group from **Source Server Groups** and move the machine group to **Applied Server Groups**. Then, click **Next**.

   > **Notice** If the heartbeat status of the machine group is **FAIL**, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
6. In the Specify Data Source step, specify the data source and click **Next**.

   Configure the parameters that are used to collect stdout and stderr in the **Plug-in Config** field. Example:

```
{
    "inputs":[
        {
            "type":"service_docker_stdout",
            "detail":{
                "Stdout":true,
                "Stderr":true,
                "IncludeContainerLabel":{
                    "LabelKey":"LabelValue"
                },
                "ExcludeContainerLabel":{
                    "LabelKey":"LabelValue"
                },
                "IncludeK8sLabel":{
                    "LabelKey":"LabelValue"
                },
                "ExcludeK8sLabel":{
                    "LabelKey":"LabelValue"
                },
                "IncludeEnv":{
                    "EnvKey":"EnvValue"
                },
                "ExcludeEnv":{
                    "EnvKey":"EnvValue"
                },
                "ExternalK8sLabelTag":{
                    "EnvKey":"EnvValue"
                },
                "ExternalEnvTag":{
                    "EnvKey":"EnvValue"
                },
                "K8sNamespaceRegex":"^(default|kube-system)$",
                "K8sPodRegex":"^(deploy.*)$",
                "K8sContainerRegex":"^(container1|container2)$"
            }
        }
    ]
}
```

Configure the following parameters:

- Data source type

  The type of the data source is fixed as service_docker_stdout.

- Parameters related to container filtering
  - For versions earlier than Logtail V1.0.29, containers can be filtered only by using environment variables and container labels.

The namespace of a Kubernetes cluster and the name of a container in the Kubernetes cluster can be mapped to container labels. The value of the LabelKey parameter for the namespace is `io.kubernetes.pod.namespace`. The value of the LabelKey parameter for the container name is `io.kubernetes.container.name`. We recommend that you use the two container labels to filter containers. If the container labels do not meet your business requirements, you can use the environment variable whitelist or the environment variable blacklist to filter containers. For example, the namespace of a pod is backend-prod, and the name of a container in the pod is worker-server. If you want the logs of the worker-server container to be collected, you can specify `"io.kubernetes.pod.namespace" : "backend-prod"` or `"io.kubernetes.container.name" : "worker-server"` in the container label whitelist.

Notice

- Container labels are retrieved by running the docker inspect command. Container labels are different from Kubernetes labels. For more information, see Obtain container labels.
- Environment variables are the same as the environment variables that are configured to start containers. For more information, see Obtain environment variables.
- Do not specify duplicate values for the LabelKey parameter. If you specify duplicate values for the LabelKey parameter, only one of the values takes effect.

Parameter	Data type	Required	Description

Parameter	Data type	Required	Description
IncludeLabel	Map (The values of the LabelKey and LabelValue parameters are strings.)	No	The container label whitelist. The whitelist specifies the containers from which stdout and stderr are collected. By default, this parameter is empty, which indicates that stdout and stderr are collected from all containers. When you configure the container label whitelist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose container labels contain the keys specified by LabelKey are matched. ■ If you specify a value for the LabelValue parameter, containers whose container labels consist of the key-value pairs specified by LabelKey and LabelValue are matched. By default, the value of the LabelValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of container labels are the same as the value of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the LabelKey parameter to *io.kubernetes.container.name* and set the LabelValue parameter to *^(nginx\|cube)$*, a container named nginx and a container named cube are matched. Key-value pairs are connected by using the OR operator. If a container label consists of one of the specified key-value pairs, the container is matched.

Parameter	Data type	Required	Description
ExcludeLabel	Map (The values of the LabelKey and LabelValue parameters are strings.)	No	The container label blacklist. The blacklist specifies the containers from which stdout and stderr are not collected. By default, this parameter is empty, which indicates that stdout and stderr are collected from all containers. When you configure the container label blacklist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose container labels contain the keys specified by LabelKey are filtered out. ■ If you specify a value for the LabelValue parameter, containers whose container labels consist of the key-value pairs specified by LabelKey and LabelValue are filtered out. By default, the value of the LabelValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of container labels are the same as the value of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the LabelValue parameter, regular expression matching is performed. For example, if you set the LabelKey parameter to *io.kubernetes.container.name* and set the LabelValue parameter to *^(nginx\|cube)$*, a container named nginx and a container named cube are matched. Key-value pairs are connected by using the OR operator. If a container label consists of one of the specified key-value pairs, the container is filtered out.

Parameter	Data type	Required	Description
IncludeEnv	Map (The values of the EnvKey and EnvValue parameters are strings.)	No	The environment variable whitelist. The whitelist specifies the containers from which stdout and stderr are collected. By default, this parameter is empty, which indicates that stdout and stderr are collected from all containers. When you configure the environment variable whitelist, the EnvKey parameter is required, and the EnvValue parameter is optional. ■ If you leave the EnvValue parameter empty, containers whose environment variables contain the keys specified by EnvKey are matched. ■ If you specify a value for the EnvValue parameter, containers whose environment variables consist of the key-value pairs specified by EnvKey and EnvValue are matched. By default, the value of the EnvValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of environment variables are the same as the value of the EnvValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the EnvValue parameter, regular expression matching is performed. For example, if you set the EnvKey parameter to *NGINX_SERVICE_PORT* and set the EnvValue parameter to *^(80\|6379)$*, containers whose port number is 80 and containers whose port number is 6379 are matched. Key-value pairs are connected by using the OR operator. If an environment variable consists of one of the specified key-value pairs, the container is matched.

Parameter	Data type	Required	Description
ExcludeEnv	Map (The values of the EnvKey and EnvValue parameters are strings.)	No	The environment variable blacklist. The blacklist specifies the containers from which stdout and stderr are not collected. By default, this parameter is empty, which indicates that stdout and stderr are collected from all containers. When you configure the environment variable blacklist, the EnvKey parameter is required, and the EnvValue parameter is optional. ■ If you leave the EnvValue parameter empty, containers whose environment variables contain the keys specified by EnvKey are filtered out. ■ If you specify a value for the EnvValue parameter, containers whose environment variables consist of the key-value pairs specified by EnvKey and EnvValue are filtered out. By default, the value of the EnvValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of environment variables are the same as the value of the EnvValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`) for the EnvValue parameter, regular expression matching is performed. For example, if you set the EnvKey parameter to *NGINX_SERVICE_PORT* and set the EnvValue parameter to *^(80\|6379)$*, containers whose port number is 80 and containers whose port number is 6379 are matched. Key-value pairs are connected by using the OR operator. If an environment variable consists of one of the specified key-value pairs, the container is filtered out.

- For Logtail V1.0.29 or later, we recommend that you use different levels of Kubernetes information, such as pod names, namespaces, container names, and labels to filter containers.

> **Note** If you change Kubernetes labels when Kubernetes control resources, such as Deployment, are running, the operational pod is not restarted. Therefore, the pod cannot detect the change. This may cause a matching rule to become invalid. When you specify the Kubernetes label whitelist and the Kubernetes label blacklist, we recommend that you use the Kubernetes labels of pods. For more information about Kubernetes labels, see Labels and Selectors.

Parameter	Data type	Required	Description

Parameter	Data type	Required	Description
IncludeK8sLabel	Map (The values of the LabelKey and LabelValue parameters are strings.)	No	The Kubernetes label whitelist. The whitelist specifies the containers from which stdout and stderr are collected. When you configure the Kubernetes label whitelist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose Kubernetes labels contain the keys specified by LabelKey are matched. ■ If you specify a value for the LabelValue parameter, containers whose Kubernetes labels consist of the key-value pairs specified by LabelKey and LabelValue are matched. By default, the value of the LabelValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of Kubernetes labels are the same as the value of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`), regular expression matching is performed. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to *^(test1\|test2)$*, containers whose Kubernetes labels consist of app:test1 and app:test2 are matched. Key-value pairs are connected by using the OR operator. If a Kubernetes label consists of one of the specified key-value pairs, the container is matched.

Parameter	Data type	Required	Description
ExcludeK8sLabel	Map (The values of the LabelKey and LabelValue parameters are strings.)	No	The Kubernetes label blacklist. The blacklist specifies the containers from which stdout and stderr are not collected. When you configure the Kubernetes label blacklist, the LabelKey parameter is required, and the LabelValue parameter is optional. ■ If you leave the LabelValue parameter empty, containers whose Kubernetes labels contain the keys specified by LabelKey are filtered out. ■ If you specify a value for the LabelValue parameter, containers whose Kubernetes labels consist of the key-value pairs specified by LabelKey and LabelValue are filtered out. By default, the value of the LabelValue parameter is a string. In this case, string matching is performed. Containers are matched only if the values of Kubernetes labels are the same as the value of the LabelValue parameter. If you specify a value that starts with a caret (`^`) and ends with a dollar sign (`$`), regular expression matching is performed. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to *^(test1\|test2)$*, containers whose Kubernetes labels consist of app:test1 and app:test2 are matched. Key-value pairs are connected by using the OR operator. If a Kubernetes label consists of one of the specified key-value pairs, the container is filtered out.
K8sNamespaceRegex	string	No	The namespace. The namespace specifies the containers from which stdout and stderr are collected. Regular expression matching is supported. For example, if you specify "K8sNamespaceRegex":"^(default|nginx)$", all containers in the nginx and default namespaces are matched.

Parameter	Data type	Required	Description
K8sPodRegex	string	No	The pod name. The pod name specifies the containers from which stdout and stderr are collected. Regular expression matching is supported. For example, if you specify "K8sPodRegex":"^(nginx-log-demo.*)$",, all containers in the pod whose name starts with nginx-log-demo are matched.
K8sContainerRegex	string	No	The container name. The container name specifies the containers from which stdout and stderr are collected. Regular expression matching is supported. Kubernetes container names are defined in spec.containers. For example, if you specify "K8sContainerRegex":"^(container-test)$", all containers whose name is container-test are matched.

- Parameters related to log labels

  For Logtail V1.0.29 or later, we recommend that you specify environment variables and Kubernetes labels for logs as log labels.

Parameter	Data type	Required	Description
ExternalEnvTag	Map (The values of the EnvKey and EnvValue parameters are strings.)	No	After you specify environment variables as log labels, Log Service adds environment variable-related fields to logs. For example, if you set the **EnvKey** parameter to *VERSION* and set the **EnvValue** parameter to *env_version*, Log Service adds the __tag__:__env_version__: v1.0.0 field to logs if the environment variable configurations of a container include `VERSION=v1.0.0`.
ExternalK8sLabelTag	Map (The values of the LabelKey and LabelValue parameters are strings.)	No	After you specify Kubernetes labels as log labels, Log Service adds Kubernetes label-related fields to logs. For example, if you set the **LabelKey** parameter to *app* and set the **LabelValue** parameter to `k8s_label_app`, Log Service adds the __tag__:__k8s_label_app__: serviceA field to logs if the label configurations of a Kubernetes cluster include `app=serviceA`.

- Other parameters

Parameter	Data type	Required	Description
Stdout	boolean	No	Specifies whether to collect stdout. By default, this parameter is empty, which indicates that stdout is collected.

Parameter	Data type	Required	Description
Stderr	boolean	No	Specifies whether to collect stderr. By default, this parameter is empty, which indicates that stderr is collected.
BeginLineRegex	string	No	The regular expression that is used to match the beginning of the first line of a log. By default, this parameter is empty, which indicates that each line is regarded as a log. If the beginning of a line matches the specified regular expression, the line is regarded as the first line of a new log. If the beginning of a line does not match the specified regular expression, the line is regarded as a part of the last log.
BeginLineTimeoutMs	int	No	The timeout period for matching the beginning of the first line of a log based on the specified regular expression. By default, this parameter is empty, which indicates that the timeout period is 3,000 milliseconds. If no new log is generated within 3,000 milliseconds, Logtail stops matching the beginning of the first line of a log and uploads the last log to Log Service.
BeginLineCheckLength	int	No	The size of the beginning of the first line of a log that matches the specified regular expression. By default, this parameter is empty, which indicates that the size of the beginning of the first line of a log is 10,240 bytes. You can configure this parameter to check whether the beginning of the first line of a log matches the specified regular expression. We recommend that you configure this parameter to improve the match efficiency.
MaxLogSize	int	No	The maximum size of a log. By default, this parameter is empty, which indicates that the maximum size of a log is 524,288 bytes. If the size of a log exceeds the value of this parameter, Logtail stops matching the beginning of the first line of a log and uploads the log to Log Service.

Parameter	Data type	Required	Description
StartLogMaxOffset	int	No	The maximum size of historical data that can be traced the first time Logtail collects logs from a log file. Valid values: [131072,1048576]. Unit: bytes. By default, this parameter is empty. In this case, the maximum size of historical data that can be traced is 131,072 bytes, which is equivalent to 128 KB.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

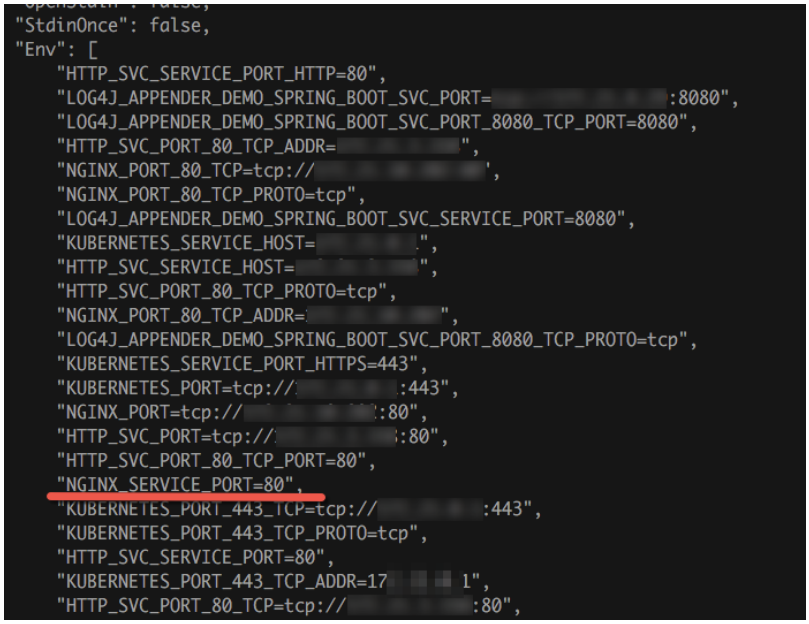
Examples of Logtail configurations for single-line logs

Example 1: Filter containers based on the environment variable whitelist and the environment variable blacklist

Collect stdout and stderr from the containers whose environment variable configurations include `NGINX_SERVICE_PORT=80` but exclude `POD_NAMESPACE=kube-system`.

1. Obtain environment variables.

   To view the environment variables of a container, you can log on to the host on which the container resides. For more information, see Obtain environment variables.



2. Create a Logtail configuration.

   Example:

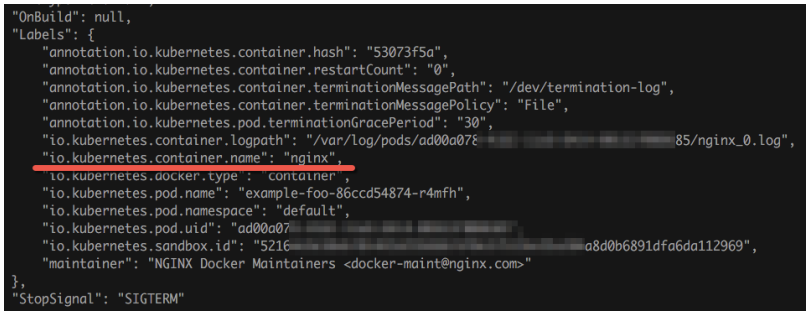
```
{
    "inputs": [
        {
            "type": "service_docker_stdout",
            "detail": {
                "Stdout": true,
                "Stderr": true,
                "IncludeEnv": {
                    "NGINX_SERVICE_PORT": "80"
                },
                "ExcludeEnv": {
                    "POD_NAMESPACE": "kube-system"
                }
            }
        }
    ]
}
```

Example 2: Filter containers based on the container label whitelist and the container label blacklist

Collect stdout and stderr from the containers whose container label is `io.kubernetes.container.name=nginx`.

1. Obtain container labels.

   To view the labels of a container, you can log on to the host on which the container resides. For more information, see Obtain container labels.

2. Create a Logtail configuration.

   Example:

```
{
    "inputs": [
        {
            "type": "service_docker_stdout",
            "detail": {
                "Stdout": true,
                "Stderr": true,
                "IncludeLabel": {
                    "io.kubernetes.container.name": "nginx"
                }
            }
        }
    ]
}
```

Example 3: Filter containers by using Kubernetes namespaces, pod names, and container names

Collect stdout and stderr from the nginx-log-demo-0 container in pods whose name starts with nginx-log-demo in the default namespace.

1. Obtain different levels of Kubernetes information.
   i. Obtain information about pods.

```
~/.kube » kubectl get pods
NAME                      READY   STATUS    RESTARTS   AGE
nginx-log-demo-0-bxl79    1/1     Running   0          48d
nginx-log-demo-1-qmrqk    1/1     Running   0          48d
nginx-log-demo-2-7khv9    1/1     Running   0          48d
nginx-log-demo-3-j24xc    1/1     Running   0          48d
```

   ii. Obtain information about namespaces.

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubernetes.io/psp: ack.privileged
  creationTimestamp: "2022-01-06T18:42:43Z"
  generateName: nginx-log-demo-0-
  labels:
    controller-uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
    job-name: nginx-log-demo-0
  name: nginx-log-demo-0-bxl79
  namespace: default
  ownerReferences:
  - apiVersion: batch/v1
    blockOwnerDeletion: true
    controller: true
    kind: Job
    name: nginx-log-demo-0
    uid: ae3eedc4-1667-458b-a6fe-39888576dbf4
  resourceVersion: "50566856"
  uid: ee10fb7d-d989-47b3-bc2a-e9ffbe767849
spec:
  containers:
  - args:
    - --log-type=nginx
    - --stdout=true
    - --total-count=100000000000
    - --log-file-size=1000000000
    - --log-file-count=2
    - --logs-per-sec=2778
    command:
    - /bin/mock_log
    image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
    imagePullPolicy: Always
    name: nginx-log-demo-0
```

2. Create a Logtail configuration.

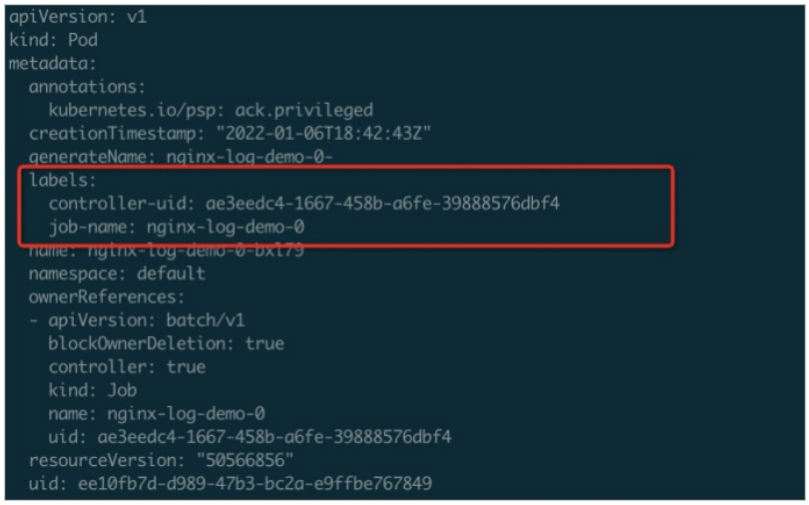
Example:

```
{
    "inputs": [
        {
            "type": "service_docker_stdout",
            "detail": {
                "Stdout": true,
                "Stderr": true,
                "K8sNamespaceRegex":"^(default)$",
                "K8sPodRegex":"^(nginx-log-demo.*)$",
                "K8sContainerRegex":"^(nginx-log-demo-0)$"
            }
        }
    ]
}
```

Example 4: Filter containers by using Kubernetes labels

Collect stdout and stderr from containers whose Kubernetes labels contain the job-name key and a specific value. The value starts with nginx-log-demo.

1. Obtain Kubernetes labels.



2. Create a Logtail configuration.

   Example:

```
{
    "inputs": [
        {
            "type": "service_docker_stdout",
            "detail": {
                "Stdout": true,
                "Stderr": true,
                "IncludeK8sLabel":{
                    "job-name":"^(nginx-log-demo.*)$"
                }
            }
        }
    ]
}
```

Examples of Logtail configurations for multi-line logs

Java exception stack logs are multi-line logs. You can create a Logtail configuration to collect the Java exception stack logs based on the following descriptions:

- Sample logs

```
2021-02-03 14:18:41.968  INFO [spring-cloud-monitor] [nio-8080-exec-4] c.g.s.web.controll
er.DemoController : service start
2021-02-03 14:18:41.969 ERROR [spring-cloud-monitor] [nio-8080-exec-4] c.g.s.web.controll
er.DemoController : java.lang.NullPointerException
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChai
n.java:193)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:1
66)
at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
...
2021-02-03 14:18:41.968  INFO [spring-cloud-monitor] [nio-8080-exec-4] c.g.s.web.controll
er.DemoController : service start done
```

- Logtail configuration

  Collect the Java exception stack logs of the containers whose container label is `app=monitor`. The Java exception stack logs start with a date that is in a fixed format. Logtail matches only the first 10 bytes of each line to improve match efficiency. After the logs are collected to Log Service, Log Service uses regular expressions to parse the logs into fields such as time, level, module, thread, and message.

  - inputs is required and is used to configure the log collection settings for the Logtail configuration. You must configure inputs based on your data source.

    > **Note** You can configure only one type of data source in inputs.

  - processors is optional and is used to configure the log processing settings for the Logtail configuration. You can specify one or more processing methods. For more information, see Use Logtail plug-ins to process data.

```
{
"inputs": [
  {
    "detail": {
      "BeginLineCheckLength": 10,
      "BeginLineRegex": "\\d+-\\d+-\\d+.*",
      "IncludeLabel": {
        "app": "monitor"
      }
    },
    "type": "service_docker_stdout"
  }
],
"processors": [
    {
        "type": "processor_regex",
        "detail": {
            "SourceKey": "content",
            "Regex": "(\\d+-\\d+-\\d+ \\d+:\\d+:\\d+\\.\\d+)\\s+(\\w+)\\s+\\[([^]]+)]\\s+
\\[([^]]+)]\\s+([\\s\\S]*)",
            "Keys": [
                "time",
                "level",
                "module",
                "thread",
                "message"
            ],
            "NoKeyError": true,
            "NoMatchError": true,
            "KeepSource": false
        }
    }
]
}
```

- Parsed logs

  For example, if the collected log is `2018-02-03 14:18:41.968 INFO [spring-cloud-monitor] [nio-80 80-exec-4] c.g.s.web.controller.DemoController : service start done`, the log is parsed into the following fields:

```
__tag__:__hostname__:logtail-dfgef
_container_name_:monitor
_image_name_:example.com-hangzhou.aliyuncs.xxxxxxxxxxxxxx
_namespace_:default
_pod_name_:monitor-6f54bd5d74-rtzc7
_pod_uid_:7f012b72-04c7-11e8-84aa-00163f00c369
_source_:stdout
_time_:2018-02-02T14:18:41.979147844Z
time:2018-02-02 02:18:41.968
level:INFO
module:spring-cloud-monitor
thread:nio-8080-exec-4
class:c.g.s.web.controller.DemoController
message:service start done
```

Log fields

The following table describes the fields that are uploaded by default for each log in a Kubernetes cluster.

Log field	Description
time	The time at which the data is uploaded. Example: `2021-02-02T02:18:41.979147844Z`.
source	The type of the data source. Valid values: stdout and stderr.
_image_name_	The name of the image.
_container_name_	The name of the container.
_pod_name_	The name of the pod.
namespace	The namespace of the pod.
_pod_uid_	The unique identifier of the pod.

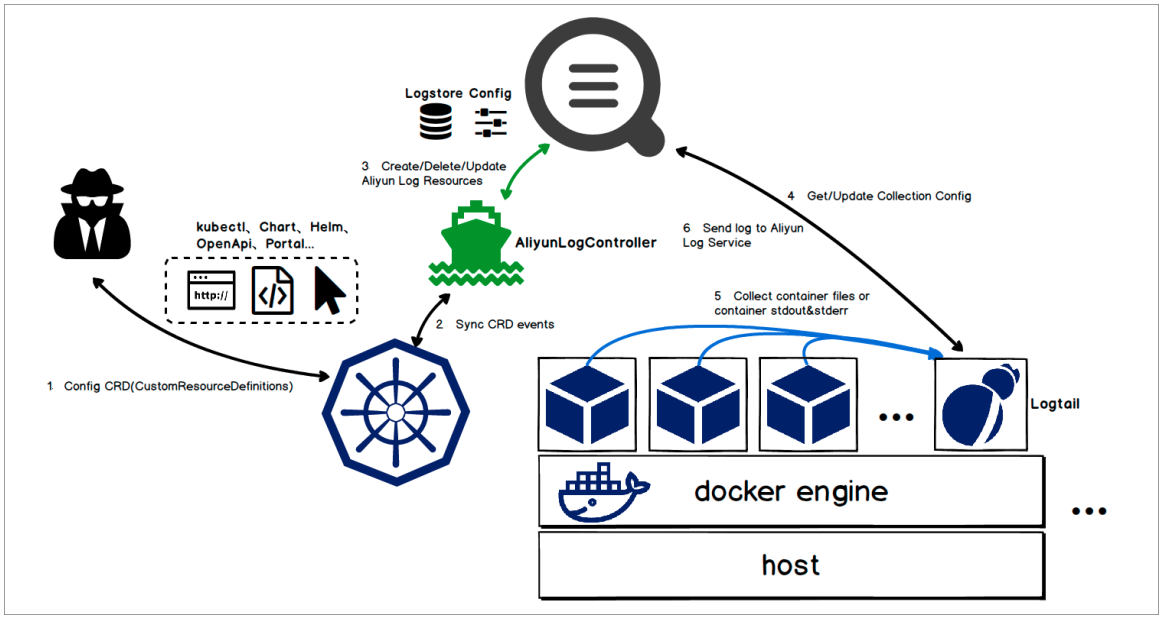
3.6.5. Use CRDs to collect container logs in DaemonSet mode

After you install Logtail in DaemonSet mode in a container, you can use a custom resource definition (CRD) to create a Logtail configuration and use the Logtail configuration to collect container logs.

Prerequisites

The Logtail component is installed. For more information, see Install Logtail components.

Implementation



The following list describes the process in which logs are collected by using a CRD:

1. The `kubectl` tool or other tools are used to create an AliyunLogConfig CRD.
2. The alibaba-log-controller detects that the CRD is updated.
3. The alibaba-log-controller sends requests to Log Service to create a Logstore, create a Logtail configuration, and apply the Logtail configuration to a machine group based on the content of the CRD and the status of the Logtail configuration in Log Service.
4. Logtail periodically sends a request to the server on which the Logtail configuration is created to obtain the new or updated Logtail configuration and perform hot reloading.
5. Logtail collects stdout and stderr logs or text logs from each container based on the obtained Logtail configuration.
6. Logtail sends the collected container logs to Log Service.

Limits

- Limits on text log collection
  - If Logtail detects the `die` event on a container that is stopped, Logtail no longer collects text logs from the container. If collection latency occurs, some text logs that are generated before the container is stopped may be lost.
  - Logtail cannot access the symbolic link of a container. You must specify an actual path as the collection directory.
  - If a volume is mounted to the data directory of a container, Logtail cannot collect data from the parent directory of the data directory. You must specify the complete path of the data directory as the collection directory.

    For example, if a volume is mounted to the */var/log/service* directory and you set the collection directory to */var/log*, Logtail cannot collect logs from the /var/log directory. You must specify */var/log/service* as the collection directory.

- By default, Kubernetes mounts the root directory of the host to the `/logtail_host` directory of the Logtail container. If you want to collect text logs from the host, you must specify `/logtail_host` as the prefix of the log file path.

  For example, if you want to collect logs from the `/home/logs/app_log/` directory of the host, you must specify `/logtail_host/home/logs/app_log/` as the log file path.

- For Docker containers, only overlay and overlay2 storage drivers are supported. If other storage drivers are used, you must mount a volume to the directory of logs. Then, a temporary directory is generated.

- Limits on stdout and stderr log collection

  The logging driver collects stdout and stderr logs only in the JSON format from containers that use the Docker engine.

- General limits

  Logtail collects data from containers that use the Docker engine or containerd engine.

  - Docker: Logtail accesses the Docker engine in the */run/docker.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.
  - containerd: Logtail accesses the containerd engine in the */run/containerd/containerd.sock* directory. Make sure that the directory exists and Logtail has the permissions to access the directory.

Create a Logtail configuration

To create a Logtail configuration, you need to only create an AliyunLogConfig CRD. After the Logtail configuration is created, Logtail automatically collects logs to Log Service based on the Logtail configuration. If you want to delete the Logtail configuration, you need to only delete the CRD.

1. Log on to your Kubernetes cluster.
2. Run the following command to create a YAML file.

   In this example, the file name is *cube.yaml*. You can specify a file name based on your business requirements.

   ```
   vim cube.yaml
   ```

3. Enter the following script in the YAML file and configure the parameters based on your business requirements.

   > **Notice**
   >
   > - The value of the configName parameter must be unique in the Log Service project that you use.
   > - If multiple CRDs are associated with the same Logtail configuration, the Logtail configuration is affected when you delete or modify one of the CRDs. After the deletion or modification, the status of the other CRDs that are associated with the Logtail configuration becomes inconsistent with the status of the Logtail configuration in Log Service.

```
apiVersion: log.alibabacloud.com/v1alpha1      # The default value is used. You do not
need to modify this parameter.
kind: AliyunLogConfig                          # The default value is used. You do not
need to modify this parameter.
metadata:
  name: simple-stdout-example                  # The name of the resource. The name mus
t be unique in the current Kubernetes cluster.
spec:
  project: k8s-my-project                      # Optional. The name of the project. The
default value is the name of the project that you use to install the Logtail component.

  logstore: k8s-stdout                         # The name of the Logstore. If the Logst
ore that you specify does not exist, Log Service automatically creates a Logstore.
  shardCount: 2                                # Optional. The number of shards. Valid
values: 1 to 10. Default value: 2.
  lifeCycle: 90                                # Optional. The data retention period fo
r the Logstore. The value of this parameter takes effect only when you create a Logstor
e. Valid values: 1 to 3650. Unit: days. Default value: 90. A value of 3650 specifies th
at log data is permanently stored in the Logstore.
  logtailConfig:                               # The Logtail configuration.
    inputType: plugin                          # The type of the data source. Valid val
ues: file and plugin. A value of file specifies text logs. A value of plugin specifies
stdout and stderr logs.
    configName: simple-stdout-example          # The name of the Logtail configuration.
The name must be the same as the resource name that is specified in metadata.name.
    inputDetail:                               # The detailed settings of the Logtail c
onfiguration. For more information, see the following configuration examples.
      ...
```

Parameter	Data type	Required	Description
project	string	No	The name of the project. The default value is the name of the project that you use to install the Logtail component.
logstore	string	Yes	The name of the Logstore. If the Logstore that you specify does not exist, Log Service automatically creates a Logstore.
shardCount	int	No	The number of shards. Valid values: 1 to 10. Default value: 2.

Parameter	Data type	Required	Description
lifeCycle	int	No	The data retention period for the Logstore. Valid values: 1 to 3650. Unit: days. Default value: 90. A value of 3650 specifies that log data is permanently stored in the Logstore. **Notice** The value of this parameter takes effect only when you create a Logstore. If you change the value of the lifeCycle parameter for an existing Logstore that is specified by the logstore parameter, the new value does not take effect.
machineGroups	array	No	The machine group to which the Logtail configuration is applied. The default value is the machine group named `k8s-group-${your_k8s_cluster_id}`. This machine group is automatically created by Log Service when you install the Logtail component.
logtailConfig	object	Yes	The detailed settings of the Logtail configuration. In most cases, you need to only configure the inputType, configName, and inputDetail parameters. For more information about the parameters, see Logtail configurations. For more information about configuration examples, see Examples of Logtail configurations that are used to collect stdout and stderr logs and Examples of Logtail configurations that are used to collect text logs.

4. Run the following command to apply the Logtail configuration.

   In this example, the file name is *cube.yaml*. You can specify a file name based on your business requirements.

   ```
   kubectl apply -f cube.yaml
   ```

   After the Logtail configuration is applied, Logtail collects stdout and stderr logs or text logs from each container, and then sends the collected logs to Log Service.

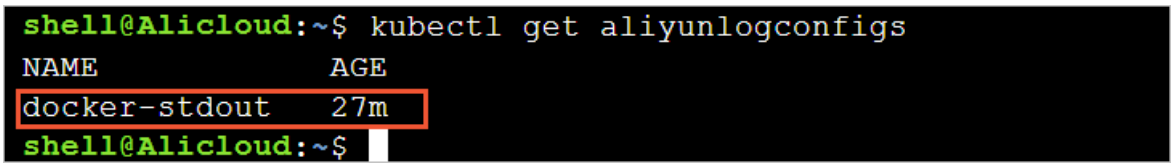
View Logtail configurations

You can view Logtail configurations in the Log Service console or by using CRDs. For more information about how to view Logtail configurations in the Log Service console, see View Logtail configurations.

> **Notice** If you modify the settings of a Logtail configuration in the Log Service console and you view the Logtail configuration by using a CRD, the modification is not displayed in the returned result of the CRD. If you modify the settings of a Logtail configuration by using a CRD and you view the Logtail configuration in the Log Service console, the modification is displayed in the Log Service console.

View all Logtail configurations in the current Kubernetes cluster

You can run the `kubectl get aliyunlogconfigs` command to view all Logtail configurations. The following figure shows the result.

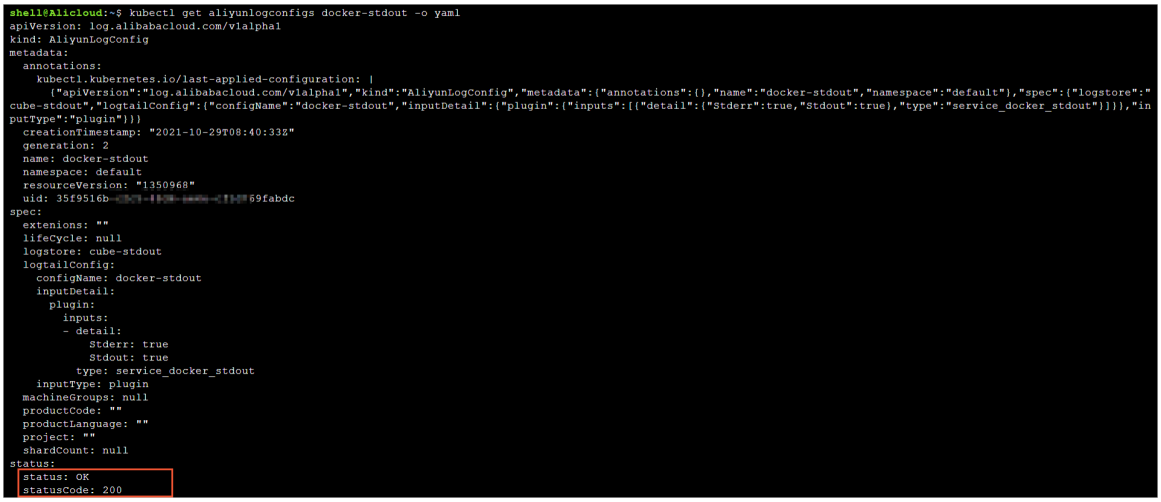


View the details and status of a Logtail configuration

You can run the `kubectl get aliyunlogconfigs config_name -o yaml` command to view the details and status of a Logtail configuration. The *config_name* field in the command specifies the name of the Logtail configuration that you want to view. You can specify a name based on your business requirements. The following figure shows the result.

The status and statusCode parameters in the result indicate the status of the Logtail configuration.

- If the value of the statusCode parameter is 200, the Logtail configuration is applied.
- If the value of the statusCode parameter is not 200, the Logtail configuration fails to be applied.



Examples of Logtail configurations that are used to collect stdout and stderr logs

If you want to collect container stdout and stderr logs, you must set the inputType parameter to `plugin` and add detailed settings to the `plugin` field of the `inputDetail` parameter. For more information about the parameters and the descriptions of the parameters, see Use the Log Service console to collect container stdout and stderr in DaemonSet mode.

Example 1: Collect container stdout and stderr logs in simple mode

Collect stdout and stderr logs from all containers except the containers whose environment variable configurations include `COLLECT_STDOUT_FLAG=false`. To view the environment variables of a container, you can log on to the host on which the container resides. For more information, see Obtain environment variables. CRD configuration example:

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: simple-stdout-example
spec:
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Service
automatically creates a Logstore.
  logstore: k8s-stdout
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. If you want to collect stdout and stderr logs, you must
set the value to plugin.
    inputType: plugin
    # The name of the Logtail configuration. The name must be the same as the resource name
that is specified in metadata.name.
    configName: simple-stdout-example
    inputDetail:
      plugin:
        inputs:
          -
            # input type
            type: service_docker_stdout
            detail:
              # The settings that allow Logtail to collect stdout and stderr logs.
              Stdout: true
              Stderr: true
              # The environment variable blacklist. In this example, stdout and stderr logs
are collected from all containers except the containers whose environment variable configur
ations include COLLECT_STDOUT_FLAG=false.
              ExcludeEnv:
                COLLECT_STDOUT_FLAG: "false"
```

Example 2: Collect container stdout logs in simple mode and process the logs by using regular expressions

To view the environment variables of a container, you can log on to the host on which the container resides. For more information, see Obtain environment variables.

Collect the access logs of Grafana from containers in simple mode and parse the access logs into structured data by using regular expressions. The environment variable configurations of the container where Grafana resides include `GF_INSTALL_PLUGINS=grafana-piechart-....`. To view the environment variables of the container, you can log on to the host on which the container resides. For more information, see Obtain environment variables.

- CRD configuration

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
```

```
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: regex-stdout-example
spec:
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Servic
e automatically creates a Logstore.
  logstore: k8s-stdout-regex
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. If you want to collect stdout logs, you must set the v
alue to plugin.
    inputType: plugin
    # The name of the Logtail configuration. The name must be the same as the resource na
me that is specified in metadata.name.
    configName: regex-stdout-example
    inputDetail:
      plugin:
        inputs:
          -
            # input type
            type: service_docker_stdout
            detail:
              # The settings that allow Logtail to collect only stdout logs.
              Stdout: true
              Stderr: false
              # The environment variable whitelist. In this example, stdout logs are coll
ected only from containers whose environment variable configurations include a key of GF_
INSTALL_PLUGINS.
              IncludeEnv:
                GF_INSTALL_PLUGINS: ''
        processors:
          -
            # The settings that allow Logtail to parse collected stdout logs by using a r
egular expression.
            type: processor_regex
            detail:
              # The name of the source field. By default, the collected stdout logs are s
tored in the content field.
              SourceKey: content
              # The regular expression that is used to extract log content.
              Regex: 't=(\d+-\d+-\w+:\d+:\d+\+\d+) lvl=(\w+) msg="([^"]+)" logger=(\w+) u
serId=(\w+) orgId=(\w+) uname=(\S*) method=(\w+) path=(\S+) status=(\d+) remote_addr=(\S+
) time_ms=(\d+) size=(\d+) referer=(\S*).*'
              # The keys that you want to extract from logs.
              Keys: ['time', 'level', 'message', 'logger', 'userId', 'orgId', 'uname', 'm
ethod', 'path', 'status', 'remote_addr', 'time_ms', 'size', 'referer']
              # The settings that allow Logtail to retain the source field.
              KeepSource: true
              # The settings that allow Logtail to report an error when the specified sou
rce field does not exist.
              NoKeyError: true
              # The settings that allow Logtail to report an error when the specified reg
ular expression does not match the value of the specified source field.
              NoMatchError: true
```

- Raw log

```
t=2018-03-09T07:14:03+0000 lvl=info msg="Request Completed" logger=context userId=0 orgId
=0 uname= method=GET path=/ status=302 remote_addr=172.16.64.154 time_ms=0 size=29 refere
r=
```

- Parsed log

Examples of Logtail configurations that are used to collect text logs

If you want to collect container text logs, you must set the `inputType` parameter to `file` and add detailed settings to the `inputDetail` parameter. For more information about the parameters and the descriptions of the parameters, see Use the Log Service console to collect container text logs in DaemonSet mode.

Example 1: Collect container text logs in simple mode

Collect container text logs whose environment variable configurations include a key of `ALIYUN_LOGTAIL_USER_DEFINED_ID`. The log file path is */data/logs/app_1/simple.LOG*.

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: simple-file-example
spec:
   # The name of the Logstore. If the Logstore that you specify does not exist, Log Service
automatically creates a Logstore.
  logstore: k8s-file
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. If you want to collect text logs, you must set the value
to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource name
that is specified in metadata.name.
    configName: simple-file-example
    inputDetail:
      # The settings that allow Logtail to collect text logs in simple mode.
      logType: common_reg_log
      # The log file path.
      logPath: /data/logs/app_1
      # The log file name. You can use wildcard characters such as asterisks (*) and questi
on marks (?) when you specify the log file name. Example: log_*.log.
      filePattern: simple.LOG
      # If you want to collect container text logs, you must set dockerFile to true.
      dockerFile: true
      # The environment variable whitelist. In this example, text logs are collected only f
rom containers whose environment variable configurations include a key of ALIYUN_LOGTAIL_US
ER_DEFINED_ID.
      dockerIncludeEnv:
        ALIYUN_LOGTAIL_USER_DEFINED_ID: ""
```

Example 2: Collect container text logs in full regex mode

A Java program generates a multi-line log that contains error stack information. You can collect the log in full regex mode and specify a regular expression that is used to match the beginning of the first line of the log in the Logtail configuration.

- Sample log

```
[2018-05-11T20:10:16,000] [INFO] [SessionTracker] [SessionTrackerImpl.java:148] Expiring
sessions
java.sql.SQLException: Incorrect string value: '\xF0\x9F\x8E\x8F",...' for column 'data'
at row 1
at org.springframework.jdbc.support.AbstractFallbackSQLExceptionTranslator.translate(Abst
ractFallbackSQLExceptionTranslator.java:84)
at org.springframework.jdbc.support.AbstractFallbackSQLException
```

- CRD configuration

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: regex-file-example
spec:
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Servic
e automatically creates a Logstore.
  logstore: k8s-file
  logtailConfig:
    # The type of the data source. If you want to collect text logs, you must set the val
ue to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource na
me that is specified in metadata.name.
    configName: regex-file-example
    inputDetail:
      # The settings that allow Logtail to collect text logs in full regex mode.
      logType: common_reg_log
      # The log file path.
      logPath: /app/logs
      # The log file name. You can use wildcard characters such as asterisks (*) and ques
tion marks (?) when you specify the log file name. Example: log_*.log.
      filePattern: error.LOG
      # The regular expression that is used to match the beginning of the first line of t
he log.
      logBeginRegex: '\[\d+-\d+-\w+:\d+:\d+,\d+]\s\[\w+]\s.*'
      # The regular expression that is used to extract log content.
      regex: '\[([^]]+)]\s\[(\w+)]\s\[(\w+)]\s\[([^:]+):(\d+)]\s(.*)'
      # The keys that you want to extract from logs.
      key : ["time", "level", "method", "file", "line", "message"]
      # The format of the time values that are extracted from logs. When logs are collect
ed in full regex mode, the time values are extracted from the time field of the logs by d
efault. If you do not want to extract time values, you can leave this parameter empty. If
you configure the timeFormat parameter, you must also configure the adjustTimezone and lo
gTimezone parameters.
      timeFormat: '%Y-%m-%dT%H:%M:%S'
      # By default, Logtail uses UTC. You must configure the following parameter before y
ou can forcefully change the time zone:
      adjustTimezone: true
      # The time zone offset. The time zone of logs is UTC+8. You can change the value of
this parameter to change the time zone.
      logTimezone: "GMT+08:00"
      # The settings that allow Logtail to upload raw logs if the logs fail to be parsed.

      discardUnmatch: false
      # If you want to collect container text logs, you must set dockerFile to true.
      dockerFile: true
      # The environment variable whitelist. In this example, text logs are collected only
from containers whose environment variable configurations include a key of ALIYUN_LOGTAIL
_USER_DEFINED_ID.
      dockerIncludeEnv:
        ALIYUN_LOGTAIL_USER_DEFINED_ID: ""
```

- Collected log

Example 3: Collect container text logs in delimiter mode

If the container text logs that you want to collect contain delimiters, you can collect the container text logs in delimiter mode. Logs that are in the delimiter-separated values (DSV) format use line feeds as boundaries. Each log is placed in a separate line. Each log is parsed into multiple fields by using delimiters.

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: delimiter-file-example
spec:
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Service
automatically creates a Logstore.
  logstore: k8s-file
  logtailConfig:
    # The type of the data source. If you want to collect text logs, you must set the value
to file.
    inputType: file
    configName: delimiter-file-example
    # The name of the Logtail configuration. The name must be the same as the resource name
that is specified in metadata.name.
    inputDetail:
      # The settings that allow Logtail to collect text logs in delimiter mode.
      logType: delimiter_log
      # The log file path.
      logPath: /usr/local/ilogtail
      # The log file name. You can use wildcard characters such as asterisks (*) and questi
on marks (?) when you specify the log file name. Example: log_*.log.
      filePattern: delimiter_log.LOG
      # The delimiter.
      separator: '|&|'
      # The keys that you want to extract from logs.
      key : ['time', 'level', 'method', 'file', 'line', 'message']
      # The name of the field from which time values are extracted.
      timeKey: 'time'
      # The format of the time values that are extracted from logs. When logs are collected
in delimiter mode, the time values are extracted from the time field of the logs by default
. If you do not want to extract time values, you can leave this parameter empty. If you con
figure the timeFormat parameter, you must also configure the adjustTimezone and logTimezone
parameters.
      timeFormat: '%Y-%m-%dT%H:%M:%S'
      # By default, Logtail uses UTC. You must configure the following parameter before you
can forcefully change the time zone:
      adjustTimezone: true
      # The time zone offset. The time zone of logs is UTC+8. You can change the value of t
his parameter to change the time zone.
      logTimezone: "GMT+08:00"
      # The settings that allow Logtail to upload raw logs if the logs fail to be parsed.
      discardUnmatch: false
      # If you want to collect container text logs, you must set dockerFile to true.
      dockerFile: true
      # The environment variable whitelist. In this example, text logs are collected only f
rom containers whose environment variable configurations include a key of ALIYUN_LOGTAIL_US
ER_DEFINED_ID.
      dockerIncludeEnv:
        ALIYUN_LOGTAIL_USER_DEFINED_ID: ''
```

Example 4: Collect container text logs in JSON mode

If the container text logs that you want to collect are JSON logs of the Object type, you can collect the container text logs in JSON mode.

- Raw log

```
{"url": "POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=U0Ujpek********&Date=Fri%2C
%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=pD12XYLmGxKQ%2Bmkd6x7hAgQ7b1c
%3D HTTP/1.1", "ip": "10.200.98.220", "user-agent": "aliyun-sdk-java", "request": {"statu
s": "200", "latency": "18204"}, "time": "05/Jan/2020:13:30:28"}
```

- CRD configuration

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in the current Kubernetes cluster.
  name: json-file-example
spec:
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Servic
e automatically creates a Logstore.
  logstore: k8s-file
  logtailConfig:
    # The type of the data source. If you want to collect text logs, you must set the val
ue to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource na
me that is specified in metadata.name.
    configName: json-file-example
    inputDetail:
      # The settings that allow Logtail to collect text logs in JSON mode.
      logType: json_log
      # The log file path.
      logPath: /usr/local/ilogtail
      # The log file name. You can use wildcard characters such as asterisks (*) and ques
tion marks (?) when you specify the log file name. Example: log_*.log.
      filePattern: json_log.LOG
      # The name of the field from which time values are extracted. If no requirements ar
e specified, set the value to timeKey: ''.
      timeKey: 'time'
      # The format of the time values that are extracted from logs. If no requirements ar
e specified, set the value to timeFormat: ''.
      timeFormat: '%Y-%m-%dT%H:%M:%S'
      # If you want to collect container text logs, you must set dockerFile to true.
      dockerFile: true
      # The environment variable whitelist. In this example, text logs are collected only
from containers whose environment variable configurations include a key of ALIYUN_LOGTAIL
_USER_DEFINED_ID.
      dockerIncludeEnv:
        ALIYUN_LOGTAIL_USER_DEFINED_ID: ""
```

3.6.6. Use CRDs to collect container text logs in Sidecar mode

This topic describes how to install Sidecar. This topic also describes how to use a custom resource definition (CRD) to create a Logtail configuration that is used to collect container text logs in Sidecar mode.

Prerequisites

The Logtail component is installed. For more information, see Install Logtail components.

Context

In Sidecar mode, the Logtail container shares a log directory with an application container. The application container writes logs to the shared directory. Logtail monitors changes to log files in the shared directory and collects logs. For more information, see Sidecar container with a logging agent and How Pods manage multiple containers.

Step 1: Install Sidecar

1. Log on to your Kubernetes cluster.
2. Create a YAML file.

   In this example, *sidecar.yaml* is used as the file name. You can specify a file name based on your business requirements.

   ```
   vim sidecar.yaml
   ```

3. Enter the following script in the YAML file and configure the parameters based on your business requirements.

   Notice Make sure that the *time zone* below *env* in the configuration file is correctly set. If the time zones are inconsistent between raw logs and processed logs in a Log Service project, the time recorded for the collected logs may be a point in time in the past or in the future. For example, if the Log Service project resides in greater China, you can set the time zone to Asia/Shanghai.

   ```
   apiVersion: batch/v1
   kind: Job
   metadata:
     name: nginx-log-sidecar-demo
     namespace: default
   spec:
     template:
       metadata:
         name: nginx-log-sidecar-demo
       spec:
         restartPolicy: Never
         containers:
         - name: nginx-log-demo
           image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
           command: ["/bin/mock_log"]
   ```

```
        args: ["--log-type=nginx", "--stdout=false", "--stderr=true", "--path=/var/log/
nginx/access.log", "--total-count=1000000000", "--logs-per-sec=100"]
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### logtail sidecar container
      - name: logtail
        # more info: https://cr.console.aliyun.com/repository/cn-hangzhou/log-service/l
ogtail/detail
        # this images is released for every region
        image: registry.cn-hangzhou.aliyuncs.com/log-service/logtail:latest
        # when recevie sigterm, logtail will delay 10 seconds and then stop
        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:
          ##### base config
          # user id
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "${your_aliyun_user_id}"
          # user defined id
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "${your_machine_group_user_defined_id}"
          # config file path in logtail's container
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/${your_region_config}/ilogtail_config.json"
          ##### env tags config
          - 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
    volumeMounts:
    - name: nginx-log
      mountPath: /var/log/nginx
  ##### share this volume
  volumes:
  - name: nginx-log
    emptyDir: {}
```

i. Configure the following basic variables in the configuration script.

```
##### base config
        # user id
        - name: "ALIYUN_LOGTAIL_USER_ID"
          value: "${your_aliyun_user_id}"
        # user defined id
        - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
          value: "${your_machine_group_user_defined_id}"
        # config file path in logtail's container
        - name: "ALIYUN_LOGTAIL_CONFIG"
          value: "/etc/ilogtail/conf/${your_region_config}/ilogtail_config.json"
```

Variable	Description
${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 of your machine group. The identifier must be unique in the region where your project resides. Example: nginx-log-sidecar. For more information, see Create a custom ID-based machine group.
${your_region_config}	Specify a value based on the ID of the region where your project resides and the type of the network for your project. For more information about regions, see Region names for Logtail installation. ■ If logs are 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 logs are 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**.

ii. Specify the mount path in the configuration script.

> **Note** We recommend that you mount a volume of the emptyDir type.

```
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### share this volume
      volumes:
      - name: nginx-log
        emptyDir: {}
```

Parameter	Description
name	The name of the volume. You can specify a name based on your business requirements. **Notice** The value of the name parameter in the volumeMounts node and the value of the name parameter in the volumes node must be the same. This ensures that the same volume is mounted for both the Logtail container and the application container.
mountPath	The mount path. You can enter the path of files in which container text logs are recorded.

iii. Specify a waiting period for the Logtail container in the configuration script.

In most cases, the waiting period is 10 seconds. This value specifies that the Logtail container exits 10 seconds after the container receives a stop command. This setting helps prevent incomplete data collection.

```
command:
- sh
- -c
- /usr/local/ilogtail/run_logtail.sh 10
```

4. Run the following command to apply the configurations in the *sidecar.yaml* file.

   In this example, *sidecar.yaml* is used as the file name. You can specify a file name based on your business requirements.

   ```
   kubectl apply -f sidecar.yaml
   ```

Step 2: Create a Logtail configuration

To create a Logtail configuration, you need only to configure an AliyunLogConfig CRD. After you create a Logtail configuration, Logtail automatically collects logs based on the Logtail configuration. If you want to delete the Logtail configuration, you need only to delete the CRD.

1. Log on to your Kubernetes cluster.
2. Run the following command to create a YAML file.

In this example, *cube.yaml* is used as the file name. You can specify a file name based on your business requirements.

```
vim cube.yaml
```

3. Enter the following script in the YAML file and configure the parameters based on your business requirements.

> **Notice**
>
> - The value of the configName parameter must be unique in the Log Service project that you use.
> - If multiple CRDs are associated with the same Logtail configuration, and you delete or modify one of the CRDs, the Logtail configuration is affected. The status of the other CRDs that are associated with the Logtail configuration becomes inconsistent with the status of the Logtail configuration in Log Service.
> - In Sidecar mode, you can collect only text logs. Therefore, you must set the dockerFile parameter to false.

```
apiVersion: log.alibabacloud.com/v1alpha1      # The default value is used. You do not
need to modify this parameter.
kind: AliyunLogConfig                          # The default value is used. You do not
need to modify this parameter.
metadata:
  name: simple-stdout-example                  # The name of the resource. The name mus
t be unique in the current Kubernetes cluster.
spec:
  project: k8s-my-project                      # Optional. The name of the project. The
default value is the name of the project that you use to install Logtail components.
  logstore: k8s-stdout                         # The name of the Logstore. If the Logst
ore that you specify does not exist, Log Service automatically creates a Logstore.
  machineGroups:                               # The name of the machine group. Set the
value to the value of ${your_machine_group_user_defined_id} when you install Sidecar. T
his machine group is used to associate Sidecar with the CRD.
  - nginx-log-sidecar
  shardCount: 2                                # Optional. The number of shards. Valid
values: 1 to 10. Default value: 2.
  lifeCycle: 90                                # Optional. The period during which log
data is stored in the Logstore. Valid values: 1 to 3650. Default value: 90. A value of
3650 specifies that data is permanently stored in the Logstore.
  logtailConfig:                               # The Logtail configuration.
    inputType: file                            # The type of the data source. In Sideca
r mode, you can use CRDs to collect only text logs. Therefore, you must set the value t
o file.
    configName: simple-stdout-example          # The name of the Logtail configuration.
The name must be the same as the resource name that is specified by the metadata.name p
arameter.
    inputDetail:                               # The detailed settings of the Logtail c
onfiguration. For more information, see the following example.
    ...
```

Parameter	Data type	Required	Description
project	string	No	The name of the project. The default value is the name of the project that you use to install Logtail components.
logstore	string	Yes	The name of the Logstore. If the Logstore that you specify does not exist, Log Service automatically creates a Logstore.
shardCount	int	No	The number of shards. Valid values: 1 to 10. Default value: 2.
lifeCycle	int	No	The period during which log data is stored in the Logstore. Valid values: 1 to 3650. Default value: 90. A value of 3650 specifies that data is permanently stored in the Logstore.

machineGroups	array	Yes	The name of the machine group. Set the value to the value of *${your_machine_group_user_defined_id}* when you install Sidecar. Example: nginx-log-sidecar.
			Log Service creates a machine group based on the name that you specify to associate Sidecar with the CRD.
			Notice
			You must specify a custom identifier for the machine group in the following format:
			`machineGroups:`
			`- nginx-log-sidecar`

logtailConfig	object	Yes	The detailed settings of the Logtail configuration. In most cases, you need only to configure the inputType, configName, and inputDetail parameters. For more information about the parameters, see Logtail configurations. For more information about configuration examples, see Configuration example for a single directory and Configuration example for different directories.

4. Run the following command to apply the Logtail configuration.

In this example, *cube.yaml* is used as the file name. You can specify a file name based on your business requirements.

```
kubectl apply -f cube.yaml
```

After you create the Logtail configuration, you can view the Logtail configuration in the Log Service console or by using a CRD. For more information, see View Logtail configurations.

Configuration example for a single directory

The following procedure shows how to use a CRD to collect text logs from the nginx-log-demo container in Sidecar mode. The container belongs to a self-managed Kubernetes cluster in a data center. The text logs include NGINX access logs and NGINX error logs and are stored in a single directory. The following list describes basic information:

- The Log Service project for log collection resides in the China (Hangzhou) region. Logs are collected over the Internet.
- The volume that needs to be mounted is nginx-log and is of the emptyDir type. The nginx-log volume will be mounted to the */var/log/nginx* directory of the nginx-log-demo and Logtail containers.
- The path to NGINX access logs is */var/log/nginx/access.log*. The Logstore that is used to store the NGINX access logs is nginx-access.
- The path to NGINX error logs is */var/log/nginx/error.log*. The Logstore that is used to store the NGINX error logs is nginx-error.
- Sidecar configuration example

```
apiVersion: batch/v1
kind: Job
metadata:
  name: nginx-log-sidecar-demo
  namespace: default
spec:
  template:
    metadata:
      name: nginx-log-sidecar-demo
    spec:
      restartPolicy: Never
      containers:
      - name: nginx-log-demo
        image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
        command: ["/bin/mock_log"]
        args: ["--log-type=nginx", "--stdout=false", "--stderr=true", "--path=/var/log/ng
inx/access.log", "--total-count=1000000000", "--logs-per-sec=100"]
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### logtail sidecar container
      - name: logtail
        # more info: https://cr.console.aliyun.com/repository/cn-hangzhou/log-service/log
tail/detail
        # this images is released for every region
        image: registry.cn-hangzhou.aliyuncs.com/log-service/logtail:latest
        # when recevie sigterm, logtail will delay 10 seconds and then stop
        command:
```

```
        command:
        - sh
        - -c
        - /usr/local/ilogtail/run_logtail.sh 10
        livenessProbe:
          exec:
            command:
            - /etc/init.d/ilogtaild
            - status
          initialDelaySeconds: 30
          periodSeconds: 30
        env:
          ##### base config
          # user id
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "1023****3423"
          # user defined id
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "nginx-log-sidecar"
          # config file path in logtail's container
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/cn-hangzhou-internet/ilogtail_config.json"
          ##### env tags config
          - 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
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### share this volume
      volumes:
      - name: nginx-log
        emptyDir: {}
```

- CRD configuration example

Create a Logtail configuration to collect NGINX access logs and another Logtail configuration to collect NGINX error logs.

- Collect NGINX access logs

> **Notice** In Sidecar mode, you must set the dockerFile parameter to false.

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in your Kubernetes cluster.
  name: nginx-log-access-example
spec:
  # The name of the project. The default value is the name of the project that you use to install Logtail components.
  project: k8s-nginx-sidecar-demo
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Service automatically creates a Logstore.
  logstore: nginx-access
  # The name of the machine group. Set the value to the value of ${your_machine_group_user_defined_id} when you install Sidecar.
  machineGroups:
  - nginx-log-sidecar
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. In Sidecar mode, you can use CRDs to collect only text logs. Therefore, you must set the value to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource name that is specified by the metadata.name parameter.
    configName: nginx-log-access-example
    inputDetail:
      # Set logType to common_reg_log.
      logType: common_reg_log
      # Specify the path to the log file.
      logPath: /var/log/nginx
      # The name of the log file. You can use wildcard characters such as asterisks (*) and question marks (?) when you specify the name of the log file. Example: log_*.log.
      filePattern: access.log
      # Set the dockerFile parameter to false. This setting is required in Sidecar mode.
      dockerFile: false
      # The regular expression that is used to match the beginning of the first line of a log. If you collect single-line logs, set the value to '.*'.
      logBeginRegex: '.*'
      # The regular expression that is used to extract log content. You can configure this parameter based on your business requirements.
      regex: '(\S+)\s(\S+)\s\S+\s\S+\s"(\S+)\s(\S+)\s+([^"]+)"\s+(\S+)\s(\S+)\s(\d+)\s(\d+)\s(\S+)\s"([^"]+)"\s.*'
      # The keys that you want to extract from logs.
      key : ["time", "ip", "method", "url", "protocol",ayloa "latency", "pd", "status", "response-size",user-agent"]
```

- Collect NGINX error logs

> **Notice** In Sidecar mode, you must set the dockerFile parameter to false.

```
# config for error log
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in your Kubernetes cluster.
  name: nginx-log-error-example
spec:
  # The name of the project. The default value is the name of the project that you use
to install Logtail components.
  project: k8s-nginx-sidecar-demo
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Serv
ice automatically creates a Logstore.
  logstore: nginx-error
  # The name of the machine group. Set the value to the value of ${your_machine_group_u
ser_defined_id} when you install Sidecar.
  machineGroups:
  - nginx-log-sidecar
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. In Sidecar mode, you can use CRDs to collect only te
xt logs. Therefore, you must set the value to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource
name that is specified by the metadata.name parameter.
    configName: nginx-log-error-example
    inputDetail:
      # Set logType to common_reg_log.
      logType: common_reg_log
      # Specify the path to the log file.
      logPath: /var/log/nginx
      # The name of the log file. You can use wildcard characters such as asterisks (*)
and question marks (?) when you specify the name of the log file. Example: log_*.log.
      filePattern: error.log
      # Set the dockerFile parameter to false. This setting is required in Sidecar mode
.
      dockerFile: false
```

Configuration example for different directories

The following procedure shows how to use a CRD to collect text logs from the nginx-log-demo container in Sidecar mode. The container belongs to a self-managed Kubernetes cluster in a data center. The text logs include NGINX access logs and are stored in different directories. The following list describes basic information:

- The Log Service project for log collection resides in the China (Hangzhou) region. Logs are collected over the Internet.
- The volumes that need to be mounted are nginx-log and nginx-logs and are of the emptyDir type. The nginx-log volume will be mounted to the */var/log/nginx* directory of the nginx-log-demo and Logtail containers. The nginx-logs volume will be mounted to the */var/log/nginxs* directory of the

nginx-log-demo and Logtail containers.

- One log file path is */var/log/nginx/access.log* and the other log file path is */var/log/nginxs/access.log*.
- The Logstore that is used to store NGINX access logs is nginx-access.
- Sidecar configuration example

```
apiVersion: batch/v1
kind: Job
metadata:
  name: nginx-log-sidecar-demo
  namespace: default
spec:
  template:
    metadata:
      name: nginx-log-sidecar-demo
    spec:
      restartPolicy: Never
      containers:
      - name: nginx-log-demo
        image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
        command: ["/bin/mock_log"]
        args: ["--log-type=nginx", "--stdout=false", "--stderr=true", "--path=/var/log/ng
inx/access.log", "--total-count=1000000000", "--logs-per-sec=100"]
        lifecycle:
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
        - name: nginx-logs
          mountPath: /var/log/nginxs
      ##### logtail sidecar container
      - name: logtail
        # more info: https://cr.console.aliyun.com/repository/cn-hangzhou/log-service/log
tail/detail
        # this images is released for every region
        image: registry.cn-hangzhou.aliyuncs.com/log-service/logtail:latest
        # when recevie sigterm, logtail will delay 10 seconds and then stop
        lifecycle:
        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:
          ##### base config
          # user id
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "1023****3423"
          # user defined id
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "nginx-log-sidecar"
          # config file path in logtail's container
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/cn-hangzhou-internet/ilogtail_config.json"
          ##### env tags config
          - 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
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
        - name: nginx-logs
          mountPath: /var/log/nginxs
      ##### share this volume
      volumes:
      - name: nginx-log
        emptyDir: {}
      - name: nginx-logs
        emptyDir: {}
```

- CRD configuration example

  Create two Logtail configurations to collect NGINX access logs from different directories.

- Collect NGINX access logs from */var/log/nginx/access.log*

> **Notice** In Sidecar mode, you must set the dockerFile parameter to false.

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in your Kubernetes cluster.
  name: nginx-log-access-example
spec:
  # The name of the project. The default value is the name of the project that you use
to install Logtail components.
  project: k8s-nginx-sidecar-demo
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Serv
ice automatically creates a Logstore.
  logstore: nginx-access
  # The name of the machine group. Set the value to the value of ${your_machine_group_u
ser_defined_id} when you install Sidecar.
  machineGroups:
  - nginx-log-sidecar
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. In Sidecar mode, you can use CRDs to collect only te
xt logs. Therefore, you must set the value to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource
name that is specified by the metadata.name parameter.
    configName: nginx-log-access-example
    inputDetail:
      # Set logType to common_reg_log.
      logType: common_reg_log
      # Specify the path to the log file.
      logPath: /var/log/nginx
      # The name of the log file. You can use wildcard characters such as asterisks (*)
and question marks (?) when you specify the name of the log file. Example: log_*.log.
      filePattern: access.log
      # Set the dockerFile parameter to false. This setting is required in Sidecar mode
.
      dockerFile: false
      # The regular expression that is used to match the beginning of the first line of
a log. If you collect single-line logs, set the value to '.*'.
      logBeginRegex: '.*'
      # The regular expression that is used to extract log content.
      regex: '(\S+)\s(\S+)\s\S+\s\S+\s"(\S+)\s(\S+)\s+([^"]+)"\s+(\S+)\s(\S+)\s(\d+)\s(
\d+)\s(\S+)\s"([^"]+)"\s.*'
      # The keys that you want to extract from logs.
      key : ["time", "ip", "method", "url", "protocol", "latency", "payload", "status",
"response-size",user-agent"]
```

- Collect NGINX access logs from */var/log/nginxs/access.log*

> **Notice** In Sidecar mode, you must set the dockerFile parameter to false.

```
apiVersion: log.alibabacloud.com/v1alpha1
kind: AliyunLogConfig
metadata:
  # The name of the resource. The name must be unique in your Kubernetes cluster.
  name: nginxs-log-access-example
spec:
  # The name of the project. The default value is the name of the project that you use
to install Logtail components.
  project: k8s-nginx-sidecar-demo
  # The name of the Logstore. If the Logstore that you specify does not exist, Log Serv
ice automatically creates a Logstore.
  logstore: nginxs-access
  # The name of the machine group. Set the value to the value of ${your_machine_group_u
ser_defined_id} when you install Sidecar.
  machineGroups:
  - nginx-log-sidecar
  # The Logtail configuration.
  logtailConfig:
    # The type of the data source. In Sidecar mode, you can use CRDs to collect only te
xt logs. Therefore, you must set the value to file.
    inputType: file
    # The name of the Logtail configuration. The name must be the same as the resource
name that is specified by the metadata.name parameter.
    configName: nginxs-log-access-example
    inputDetail:
      # Set logType to common_reg_log.
      logType: common_reg_log
      # Specify the path to the log file.
      logPath: /var/log/nginxs
      # The name of the log file. You can use wildcard characters such as asterisks (*)
and question marks (?) when you specify the name of the log file. Example: log_*.log.
      filePattern: access.log
      # Set the dockerFile parameter to false. This setting is required in Sidecar mode
.
      dockerFile: false
      # The regular expression that is used to match the beginning of the first line of
a log. If you collect single-line logs, set the value to '.*'.
      logBeginRegex: '.*'
      # The regular expression that is used to extract log content.
      regex: '(\S+)\s(\S+)\s\S+\s\S+\s"(\S+)\s(\S+)\s+([^"]+)"\s+(\S+)\s(\S+)\s(\d+)\s(
\d+)\s(\S+)\s"([^"]+)"\s.*'
      # The keys that you want to extract from logs.
      key : ["time", "ip", "method", "url", "protocol", "latency", "payload", "status",
"response-size",user-agent"]
# config for error log
```

3.6.7. Use the Log Service console to collect container text logs in Sidecar mode

This topic describes how to install Sidecar. This topic also describes how to use the Log Service console to create a Logtail configuration that is used to collect container text logs in Sidecar mode.

Prerequisites

The Logtail component is installed. For more information, see Install Logtail components.

Context

In Sidecar mode, the Logtail container shares a log directory with an application container. The application container writes logs to the shared directory. Logtail monitors changes to log files in the shared directory and collects logs. For more information, see Sidecar container with a logging agent and How Pods manage multiple containers.

Step 1: Install Sidecar

1. Log on to your Kubernetes cluster.
2. Create a YAML file.

   In this example, *sidecar.yaml* is used as the file name. You can specify a file name based on your business requirements.

   ```
   vim sidecar.yaml
   ```

3. Enter the following script in the YAML file and configure the parameters based on your business requirements.

   Notice Make sure that the *time zone* below *env* in the configuration file is correctly set. If the time zones are inconsistent between raw logs and processed logs in a Log Service project, the time recorded for the collected logs may be a point in time in the past or in the future. For example, if the Log Service project resides in greater China, you can set the time zone to Asia/Shanghai.

   ```
   apiVersion: batch/v1
   kind: Job
   metadata:
     name: nginx-log-sidecar-demo
     namespace: default
   spec:
     template:
       metadata:
         name: nginx-log-sidecar-demo
       spec:
         restartPolicy: Never
         containers:
         - name: nginx-log-demo
           image: registry.cn-hangzhou.aliyuncs.com/log-service/docker-log-test:latest
           command: ["/bin/mock_log"]
           args: ["--log-type=nginx", "--stdout=false", "--stderr=true", "--path=/var/log/
   ```

```
nginx/access.log", "--total-count=1000000000", "--logs-per-sec=100"]
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### logtail sidecar container
      - name: logtail
        # more info: https://cr.console.aliyun.com/repository/cn-hangzhou/log-service/logtail/detail
        # this images is released for every region
        image: registry.cn-hangzhou.aliyuncs.com/log-service/logtail:latest
        # when recevie sigterm, logtail will delay 10 seconds and then stop
        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:
          ##### base config
          # user id
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "${your_aliyun_user_id}"
          # user defined id
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "${your_machine_group_user_defined_id}"
          # config file path in logtail's container
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/${your_region_config}/ilogtail_config.json"
          ##### env tags config
          - 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
      volumeMounts:
      - name: nginx-log
        mountPath: /var/log/nginx
    ##### share this volume
    volumes:
    - name: nginx-log
      emptyDir: {}
```

i. Configure the following basic variables in the configuration script.

```
##### base config
          # user id
          - name: "ALIYUN_LOGTAIL_USER_ID"
            value: "${your_aliyun_user_id}"
          # user defined id
          - name: "ALIYUN_LOGTAIL_USER_DEFINED_ID"
            value: "${your_machine_group_user_defined_id}"
          # config file path in logtail's container
          - name: "ALIYUN_LOGTAIL_CONFIG"
            value: "/etc/ilogtail/conf/${your_region_config}/ilogtail_config.json"
```

Variable	Description
${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 of your machine group. The identifier must be unique in the region where your project resides. Example: nginx-log-sidecar. For more information, see Create a custom ID-based machine group.
${your_region_config}	Specify a value based on the ID of the region where your project resides and the type of the network for your project. For more information about regions, see Region names for Logtail installation. ■ If logs are 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 logs are 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**.

ii. Specify the mount path in the configuration script.

> **Note** We recommend that you mount a volume of the emptyDir type.

```
        volumeMounts:
        - name: nginx-log
          mountPath: /var/log/nginx
      ##### share this volume
      volumes:
      - name: nginx-log
        emptyDir: {}
```

Parameter	Description
name	The name of the volume. You can specify a name based on your business requirements. **Notice** The value of the name parameter in the volumeMounts node and the value of the name parameter in the volumes node must be the same. This ensures that the same volume is mounted for both the Logtail container and the application container.
mountPath	The mount path. You can enter the path of files in which container text logs are recorded.

iii. Specify a waiting period for the Logtail container in the configuration script.

In most cases, the waiting period is 10 seconds. This value specifies that the Logtail container exits 10 seconds after the container receives a stop command. This setting helps prevent incomplete data collection.

```
command:
- sh
- -c
- /usr/local/ilogtail/run_logtail.sh 10
```

4. Run the following command to apply the configurations in the *sidecar.yaml* file.

   In this example, *sidecar.yaml* is used as the file name. You can specify a file name based on your business requirements.

   ```
   kubectl apply -f sidecar.yaml
   ```

Step 2: Create a machine group

1.
2. In the Projects section, click the project that you use to install Logtail components.
3. In the left navigation sidebar, choose **Resources > Machine Groups**.
4. In the **Machine Groups** list, choose > **Create Machine Group**.

5. In the **Create Machine Group** panel, configure the following parameters and click **OK**.

Parameter	Description
Name	The name of the machine group. **Notice** After you create a machine group, you cannot change the name of the machine group. Proceed with caution.
Identifier	The identifier of the machine group. Select **Custom ID**.
Topic	The topic of the machine group. The topic is used to differentiate log data that is generated on different servers. For more information, see Log topics.
Custom Identifier	The custom identifier of the machine group. Set the value to the value of *${your_machine_group_user_defined_id}* when you install Sidecar. Example: nginx-log-sidecar.

Step 3: Create a Logtail configuration

1.
2. In the **Import Data** section, click **RegEx - Text Log**.

   This example shows how to create a Logtail configuration that is used to collect text logs in full regex mode. For more information about how to collect text logs in other modes, see Collect text logs.
3. Select a project and a Logstore. Then, click **Next**.

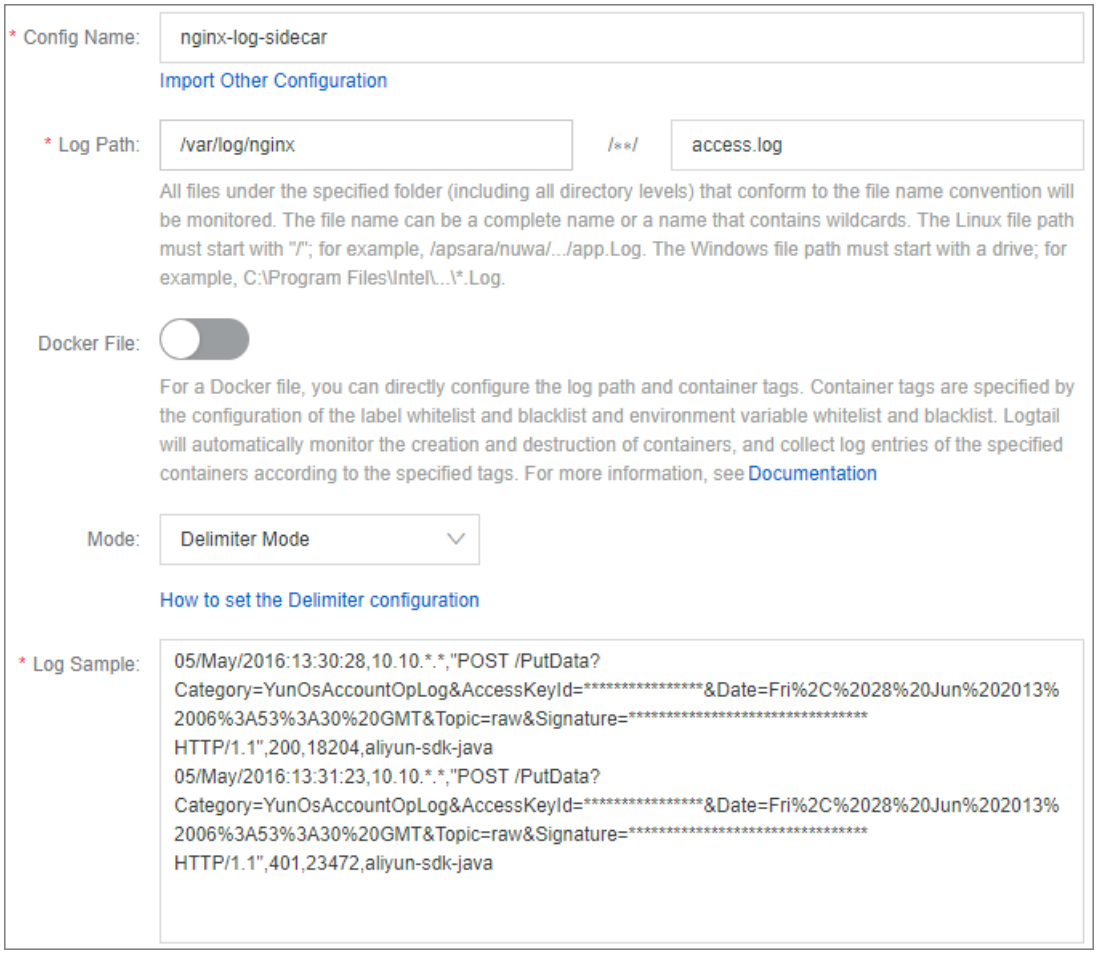
   Select the project that you use to install Logtail components and the Logstore that you create.
4. Click **Use Existing Machine Groups**.
5. Select a machine group from **Source Server Groups** and move it to **Applied Server Groups**. Then, click **Next**.

   Select the machine group that you create in Step 2: Create a machine group.

   Notice If you apply a machine group immediately after you create the machine group, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. To resolve this issue, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?
6. Create a Logtail configuration and click **Next**.

   You can collect logs in simple mode, NGINX mode, delimiter mode, JSON mode, and full regex mode. For more information, see Collect text logs.

   Notice In Sidecar mode, you must turn off **Docker File**.



7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note**
   > - If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.
   > - If a field for which indexing is enabled is of the long or double type, you cannot configure the Case Sensitive or Delimiter parameter for the field.

3.6.8. Collect logs from standard Docker containers

This topic describes how to deploy a Logtail container and create a Logtail configuration file to collect logs from standard Docker containers.

Step 1: Deploy a Logtail container

1. Run the following command to pull the Logtail image:

```
docker pull registry.cn-hangzhou.aliyuncs.com/log-service/logtail
```

Replace *registry.cn-hangzhou.aliyuncs.com* with the actual value. For more information about regions, see Region names for Logtail installation. If your server is in a virtual private cloud (VPC), you must replace registry with registry-vpc.

2. Start a Logtail container.

> **Note** Before you set the parameters, you must complete one of the following configurations. Otherwise, the `container text file busy` error may occur when you delete other containers.
>
> - For CentOS 7.4 and later, set fs.may_detach_mounts to 1. For more information, see Bug 1468249, Bug 1441737, and Issue 34538.
> - Add `--privileged` to the startup parameters to grant Logtail the `privileged` permission. For more information, see Docker run reference.

Replace the `${your_region_name}`, `${your_aliyun_user_id}`, and `${your_machine_group_user_defined_id}` parameters in the following command with the actual values:

```
docker run -d -v /:/logtail_host:ro -v /var/run:/var/run --env ALIYUN_LOGTAIL_CONFIG=/etc/ilogtail/conf/${your_region_name}/ilogtail_config.json --env ALIYUN_LOGTAIL_USER_ID=${your_aliyun_user_id} --env ALIYUN_LOGTAIL_USER_DEFINED_ID=${your_machine_group_user_defined_id} registry.cn-hangzhou.aliyuncs.com/log-service/logtail
```

Parameter	Description
`${your_region_name}`	The ID of the region where your project resides and the type of the network that your project uses. For more information about regions, see Region names for Logtail installation. ○ If your project uses the Internet, set the value in the `region-internet` format, for example, **cn-hangzhou-internet**. ○ If your project uses the Alibaba Cloud internal network, set the value in the `region` format, for example, **cn-hangzhou**.
`${your_aliyun_user_id}`	The ID of your Alibaba Cloud account. For more information, see Configure a user identifier.
`${your_machine_group_user_defined_id}`	The custom identifier of your server group. The identifier must be unique in the region where your project resides. For more information, see Create a custom ID-based machine group.

> **Note**
>
> You can customize the startup parameters of the Logtail container if the following conditions are met:
>
> i. The following environment variables are configured: `ALIYUN_LOGTAIL_USER_DEFINED_ID`, `ALIYUN_LOGTAIL_USER_ID`, and `ALIYUN_LOGTAIL_CONFIG`.
>
> ii. The */var/run* directory of the host is mounted on the */var/run* directory of the Logtail container.
>
> iii. The root directory of the host is mounted on the `/logtail_host` directory of the Logtail container.
>
> iv. If the `The parameter is invalid : uuid=none` error is returned in the */usr/local/ilogtail/ilogtail.LOG* log file, you must create a file named *product_uuid* on the host. Then, you must enter a valid UUID in the file, for example, `169E98C9-ABC0-4A92-B1D2-AA6239C0D261`, and mount the file on the */sys/class/dmi/id/product_uuid* directory of the Logtail container.

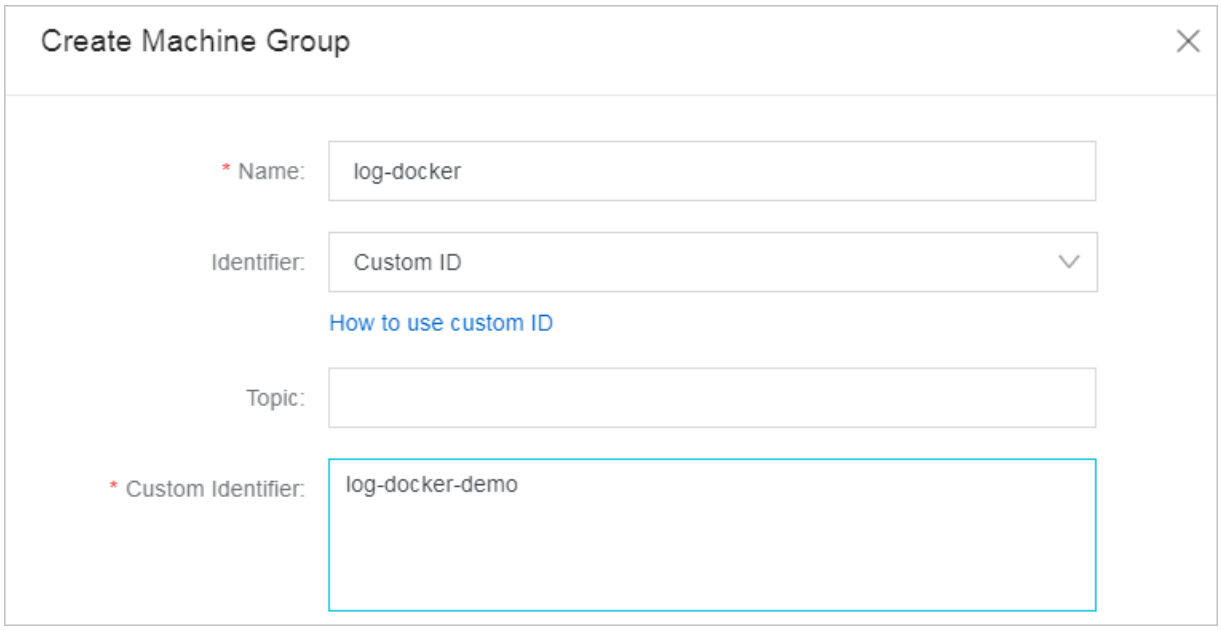
Step 2: Create a Logtail configuration file

Create a Logtail configuration file in the console based on your business requirements.

- To collect Docker text logs, follow the steps that you perform to collect Kubernetes text logs. For more information, see Use the console to collect Kubernetes text logs in DaemonSet mode.
- To collect Docker stdout and stderr logs, follow the steps that you perform to collect Kubernetes stdout and stderr logs. For more information, see Use the console to collect Kubernetes stdout and stderr logs in DaemonSet mode.
- To collect host text logs, follow the steps provided in Collect text logs.

  By default, the root directory of the host is mounted on the `/logtail_host` directory of the Logtail container. When you configure the directory to collect logs, you must add the container directory as the prefix to the log path. For example, to collect logs from the `/home/logs/app_log/` directory of the host, you must set the log path to `/logtail_host/home/logs/app_log/`.

When you create a server group, enter the value of the `ALIYUN_LOGTAIL_USER_DEFINED_ID` parameter in the **Custom Identifier** field. This value is specified in Step 1: Deploy a Logtail container.



Default fields

- Docker stdout and stderr logs

  The following table describes the fields that are uploaded by default for each log entry.

Log field	Description
`_time_`	The time when the data is uploaded, for example, `2018-02-02T02:18:41.979147844Z`.
`_source_`	The type of a data source. Valid values: stdout and stderr.
`_image_name_`	The name of an image.
`_container_name_`	The name of a container.
`_container_ip_`	The IP address assigned to the pod where a container resides.

- Docker text logs

  The following table describes the fields that are uploaded by default for each log entry.

Log field	Description
`_image_name_`	The name of an image.
`_container_name_`	The name of a container.
`_container_ip_`	The IP address assigned to the pod where a container resides.

Log field	Description

Related operations

- View the status of Logtail.

  You can run the `docker exec ${logtail_container_id} /etc/init.d/ilogtaild status` command to view the status of Logtail.

- View the version number, IP address, and startup time of Logtail.

  You can run the `docker exec ${logtail_container_id} cat /usr/local/ilogtail/app_info.json` command to view the information of Logtail.

- View the operational logs of Logtail.

  The operational logs of Logtail are stored in the `ilogtail.LOG` file in the `/usr/local/ilogtail/` directory. If the log file is rotated and compressed, it is stored as a file named `ilogtail.LOG.x.gz`.

  Example:

```
[root@iZbp17enxc2us3624wexh2Z ilogtail]# docker exec a287de895e40 tail -n 5 /usr/local/il
ogtail/ilogtail.LOG
[2018-02-06 08:13:35.721864]    [INFO]    [8]    [build/release64/sls/ilogtail/LogtailPlu
gin.cpp:104]    logtail plugin Resume:start
[2018-02-06 08:13:35.722135]    [INFO]    [8]    [build/release64/sls/ilogtail/LogtailPlu
gin.cpp:106]    logtail plugin Resume:success
[2018-02-06 08:13:35.722149]    [INFO]    [8]    [build/release64/sls/ilogtail/EventDispa
tcher.cpp:369]    start add existed check point events, size:0
[2018-02-06 08:13:35.722155]    [INFO]    [8]    [build/release64/sls/ilogtail/EventDispa
tcher.cpp:511]    add existed check point events, size:0    cache size:0    event size:0
success count:0
[2018-02-06 08:13:39.725417]    [INFO]    [8]    [build/release64/sls/ilogtail/ConfigMana
ger.cpp:3776]    check container path update flag:0    size:1
```

  The standard output of the container is irrelevant to this case. Ignore the following standard output:

```
start umount useless mount points, /shm$|/merged$|/mqueue$
umount: /logtail_host/var/lib/docker/overlay2/3fd0043af174cb0273c3c7869500fbe2bdb95d13b1e
110172ef57fe840c82155/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/d5b10aa19399992755de1f85d25009528daa749c1bf
8c16edff44beab6e69718/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/5c3125daddacedec29df72ad0c52fac800cd56c6e88
0dc4e8a640b1e16c22dbe/merged: must be superuser to unmount
......
xargs: umount: exited with status 255; aborting
umount done
start logtail
ilogtail is running
logtail status:
ilogtail is running
```

- Restart Logtail.

To restart Logtail, use the following sample code:

```
[root@iZbp17enxc2us3624wexh2Z ilogtail]# docker exec a287de895e40 /etc/init.d/ilogtaild s
top
kill process Name: ilogtail pid: 7
kill process Name: ilogtail pid: 8
stop success
[root@iZbp17enxc2us3624wexh2Z ilogtail]# docker exec a287de895e40 /etc/init.d/ilogtaild s
tart
ilogtail is running
```

3.6.9. Collect Kubernetes events

This topic describes how to use the eventer component to collect events from Kubernetes and send the events to Log Service.

Log Service allows you to collect events from Kubernetes by using the kube-eventer or K8s Event Center application.

- Kube-eventer

  For more information about the source code for Kubernetes event collection, see GitHub.

- K8s Event Center (recommended)

  To collect Kubernetes event data and configure visualized charts and alerts, you can use the **K8s Event Center** application provided in the Log Service console. For more information, see Create and use an event center.

Configure event collection

> **Note**
> - If you use Container Service for Kubernetes (ACK), see Create and use an event center.
> - If you use self-managed Kubernetes, you must set the endpoint, project, logStore, regionId, internal, accessKeyId, and accessKeySecret parameters.

The following example shows the event collection configuration:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    name: kube-eventer
  name: kube-eventer
  namespace: kube-system
spec:
  replicas: 1
  selector:
    matchLabels:
      app: kube-eventer
  template:
    metadata:
      labels:
        app: kube-eventer
```

```
      annotations:
        scheduler.alpha.kubernetes.io/critical-pod: ''
    spec:
      dnsPolicy: ClusterFirstWithHostNet
      serviceAccount: kube-eventer
      containers:
        - image: registry.cn-hangzhou.aliyuncs.com/acs/kube-eventer:v1.2.5-cc7ec54-aliyun
          name: kube-eventer
          command:
            - "/kube-eventer"
            - "--source=kubernetes:https://kubernetes.default"
            ## .send to sls
            ## --sink=sls:https://{endpoint}?project={project}&logStore=k8s-event&regionId=
{region-id}&internal=false&accessKeyId={accessKeyId}&accessKeySecret={accessKeySecret}
            - --sink=sls:https://cn-beijing.log.aliyuncs.com?project=k8s-xxxx&logStore=k8s-
event&regionId=cn-beijing&internal=false&accessKeyId=xxx&accessKeySecret=xxx
          env:
          # If TZ is assigned, set the TZ value as the time zone
          - name: TZ
            value: "Asia/Shanghai"
          volumeMounts:
            - name: localtime
              mountPath: /etc/localtime
              readOnly: true
            - name: zoneinfo
              mountPath: /usr/share/zoneinfo
              readOnly: true
          resources:
            requests:
              cpu: 10m
              memory: 50Mi
            limits:
              cpu: 500m
              memory: 250Mi
      volumes:
        - name: localtime
          hostPath:
            path: /etc/localtime
        - name: zoneinfo
          hostPath:
            path: /usr/share/zoneinfo
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: kube-eventer
rules:
  - apiGroups:
      - ""
    resources:
      - events
    verbs:
      - get
      - list
```

```
      - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: kube-eventer
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: kube-eventer
subjects:
  - kind: ServiceAccount
    name: kube-eventer
    namespace: kube-system
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: kube-eventer
  namespace: kube-system
```

Parameter	Type	Required	Description
endpoint	string	Yes	The endpoint of Log Service. For more information, see Endpoints.
project	string	Yes	The project in Log Service.
logStore	string	Yes	The Logstore in Log Service.
internal	string	Required for self-managed Kubernetes	If you use self-managed Kubernetes, set the value to false.
regionId	string	Required for self-managed Kubernetes	The region ID of Log Service. For more information, see Endpoints.
accessKeyId	string	Required for self-managed Kubernetes	The AccessKey ID. We recommend that you use the AccessKey ID of a RAM user.
accessKeySecret	string	Required for self-managed Kubernetes	The AccessKey secret. We recommend that you use the AccessKey secret of a RAM user.

Sample log entry

The following example shows a collected sample log entry:

```
hostname:  cn-hangzhou.i-***********"
level:  Normal
pod_id:  2a360760-****
pod_name:  logtail-ds-blkkr
event_id:  {
   "metadata":{
      "name":"logtail-ds-blkkr.157b7cc90de7e192",
      "namespace":"kube-system",
      "selfLink":"/api/v1/namespaces/kube-system/events/logtail-ds-blkkr.157b7cc90de7e192",
      "uid":"2aaf75ab-****",
      "resourceVersion":"6129169",
      "creationTimestamp":"2019-01-20T07:08:19Z"
   },
   "involvedObject":{
      "kind":"Pod",
      "namespace":"kube-system",
      "name":"logtail-ds-blkkr",
      "uid":"2a360760-****",
      "apiVersion":"v1",
      "resourceVersion":"6129161",
      "fieldPath":"spec.containers{logtail}"
   },
   "reason":"Started",
   "message":"Started container",
   "source":{
      "component":"kubelet",
      "host":"cn-hangzhou.i-***********"
   },
   "firstTimestamp":"2019-01-20T07:08:19Z",
   "lastTimestamp":"2019-01-20T07:08:19Z",
   "count":1,
   "type":"Normal",
   "eventTime":null,
   "reportingComponent":"",
   "reportingInstance":""
}
```

Log field	Type	Description
hostname	string	The hostname of the server where an event occurs.
level	string	The level of a log entry. Valid values: Normal and Warning.
pod_id	string	The unique identifier of a pod. This field is available only if the event type is related to the pod.
pod_name	string	The name of a pod. This field is available only if the event type is related to the pod.
eventId	json	The details of an event. The value of this field is a JSON string.

3.7. Customize Logtail plug-ins to collect data

3.7.1. Overview

This topic describes Logtail plug-ins and their configuration methods. The plug-ins are used for data collection.

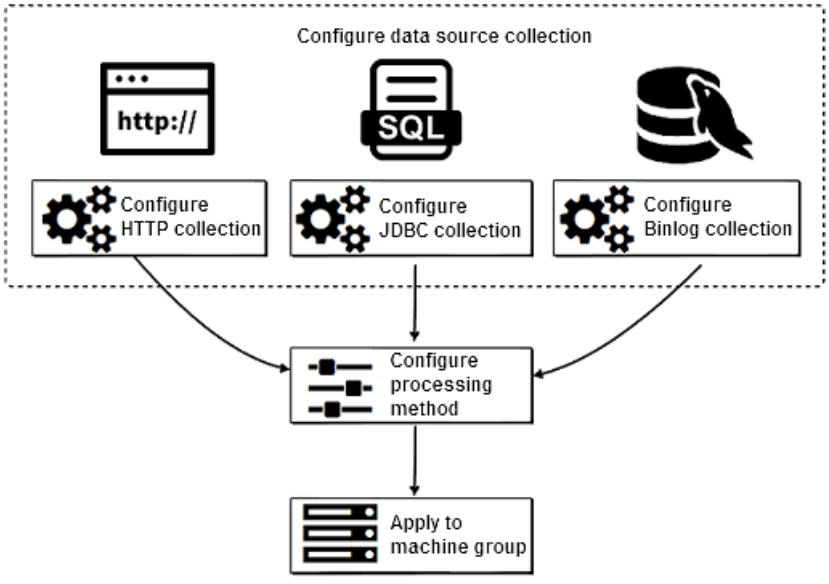
Background information

> **Notice** Logtail plug-ins do not support Linux kernel versions that are earlier than 2.6.32.

You can customize plug-ins for Logtail to collect various types of data. For example:

- You can configure Logtail plug-ins to collect HTTP data and upload processed data to Log Service. This way, you can monitor the availability of your services in real time.
- You can configure Logtail plug-ins to collect the data of MySQL databases. You can synchronize incremental data based on the auto-increment ID of the database or the time when the data is generated.
- You can configure Logtail plug-ins to collect the binary logs of MySQL databases, and search and analyze binary logs in real time.

Configuration procedure



1. Configure a collection method.

   For information about how to configure collection methods for various data sources, see the following topics:

   - Collect MySQL binary logs
   - Collect MySQL query results
   - Collect HTTP data

- Collect container standard outputs
- Collect data from Beats and Logstash
- Collect syslogs
- Collect Windows event logs
- Collect Docker events
- Collect systemd-journald logs

2. Configure a data processing method.

   You can configure multiple processing methods for a data source. Logtail processes data by using these methods in sequence. You can configure all available processing methods for each type of data source. For more information, see Overview.

3. Apply the configurations to the specified machine group.

   Apply the log collection configurations and processing configurations to the specified machine group. Then, Logtail automatically pulls the configurations and starts to collect logs.

3.7.2. Collect MySQL binary logs

This topic describes how to create a Logtail configuration in the Log Service console to collect MySQL binary logs.

Prerequisites

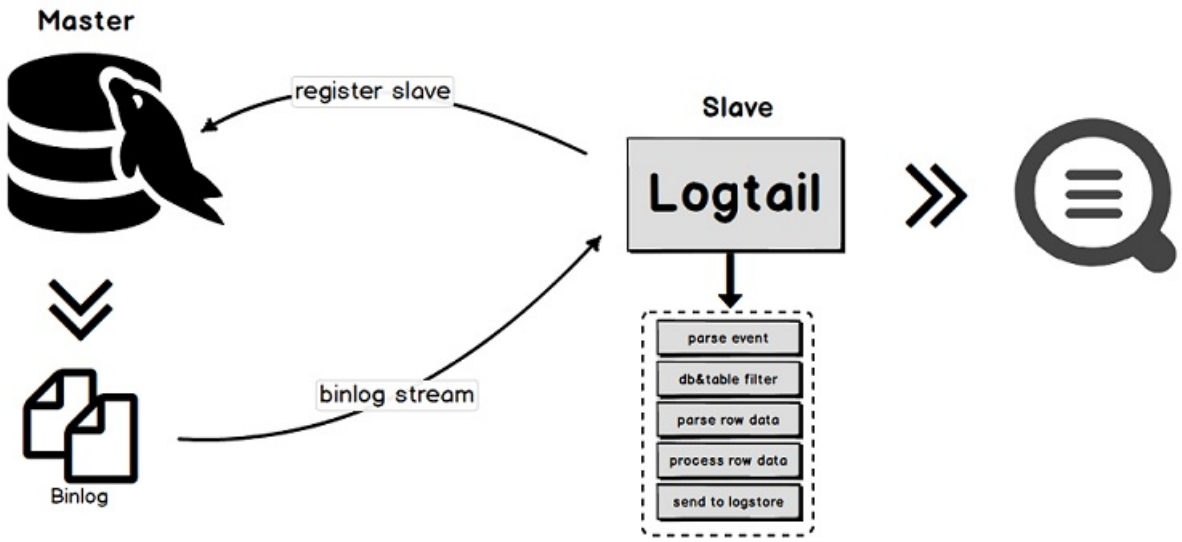
Logtail is installed on a server. For ease of understanding, this server is referred to as the Logtail server in this topic. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

> **Note** Linux servers support Logtail V0.16.0 or later. Windows servers support Logtail V1.0.0.8 or later.

Principle

Logtail acts as a slave MySQL node to communicate with the master MySQL node. The following list describes the communication process:

1. Logtail acts as a slave MySQL node and sends dump requests to the master MySQL node.
2. After the master MySQL node receives the dump requests, the node sends binary logs to Logtail in real time.
3. Logtail performs operations such as event parsing, filtering, and data parsing on the binary logs. Then, Logtail uploads the parsed data to Log Service.



Features

- Binary logs can be incrementally collected. This way, you can collect data related to the update operations that are performed on your databases. MySQL databases are supported, such as Relational Database Service (RDS) databases.
- Multiple methods are provided to allow users to filter data in databases.
- You can configure binary log checkpoints.
- Checkpoints are used to synchronize data storage status.

Limits

- Logtail V1.0.31 and later support MySQL 8.0.
- The binary logging feature must be enabled for your MySQL database, and the binlog_format parameter must be set to ROW for the database. By default, the feature is enabled for an RDS database.

```
# Check whether the binary logging feature is enabled.
mysql> show variables like "log_bin";
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| log_bin       | ON    |
+---------------+-------+
1 row in set (0.02 sec)
# View the type of the feature.
mysql> show variables like "binlog_format";
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| binlog_format | ROW   |
+---------------+-------+
1 row in set (0.03 sec)
```

- The ID of the slave MySQL node whose role Logtail assumes must be unique on the master MySQL node.

- Limits on RDS databases:
  - You cannot install Logtail on the server that hosts your RDS instance. You must install Logtail on a server that can connect to the instance.
  - You cannot collect binary logs from a secondary RDS database. You must configure your primary RDS database to collect binary logs.

Scenarios

If you want to synchronize large amounts of data and require high performance, you can collect MySQL binary logs.

- Track data changes in databases to perform real-time query and analysis.
- Audit the operations that are performed on databases.
- Use Log Service to perform operations on the update-related data of databases. For example, you can perform custom query and analysis on the data, visualize the data, push the data to downstream nodes for stream processing, import the data to MaxCompute for batch processing, and import the data to Object Storage Service (OSS) for long-term storage.

Precautions

We recommend that you relax the limits on resource usage for Logtail to accommodate traffic spikes and mitigate data risks. If the limits are exceeded, Logtail may be forced to restart.

You can modify the related parameters in the */usr/local/ilogtail/ilogtail_config.json* file. For more information, see Configure the startup parameters of Logtail.

You can relax the limit on CPU utilization to two cores and the limit on memory usage to 2,048 MB. Example:

```
{
    ...
    "cpu_usage_limit":2,
    "mem_usage_limit":2048,
    ...
}
```

Data reliability

We recommend that you enable the global transaction identifier (GTID) feature on your MySQL server and upgrade Logtail to V0.16.15 or later. This helps prevent data from being repeatedly collected after a master/slave switchover is triggered on your database and ensure data reliability.

- Incomplete data collection: If the network between Logtail and your MySQL server is disconnected for a long period of time, some data may not be collected.

  When the network between Logtail and your master MySQL node is disconnected, the master node keeps generating binary logs and deletes expired binary logs. After the network connection is re-established, Logtail uses a checkpoint from the local storage to request binary logs from the master node. However, if the network disconnection lasts long, the logs that are generated after the checkpoint may be deleted. In this case, the recovery mechanism is triggered. The mechanism identifies the most recent binary log on the master node to resume collection. The logs that are generated between the checkpoint and the most recent binary log are not collected. This leads to incomplete data collection.

- Repeated data collection: If a master/slave switchover is triggered when the sequence numbers of

binary logs are inconsistent between your master MySQL node and slave MySQL node, binary logs may be repeatedly collected.

If MySQL master/slave synchronization is configured, the master node automatically synchronizes binary logs to the slave node, which stores the logs to local binary log files. If the sequence numbers are inconsistent between the master and slave nodes and a master/slave switchover is triggered, logs may be repeatedly collected. This issue occurs because the checkpoint mechanism is based on the names of binary log files and the offsets of the files.

For example, a piece of data is in the checkpoint range from `(binlog.100, 4)` to `(binlog.105, 4)` on the master node and in the checkpoint range from `(binlog.1000, 4)` to `(binlog.1005, 4)` on the slave node. Logtail has obtained the data from the master node and updated the local checkpoint to `(binlog.105, 4)`. If a master/slave switchover is triggered and no errors occur, Logtail continues collecting binary logs from the new master node based on the local checkpoint `(binlog.105, 4)`. However, the sequence number of the data that is in the checkpoint range from `(binlog.1000, 4)` to `(binlog.1005, 4)` on the new master node is greater than the sequence number of the data that is requested by Logtail. The new master node returns all data in the range to Logtail. This leads to repeated data collection.

Procedure

1.
2. In the **Import Data** section, click **MySQL BinLog - Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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, configure **Config Name** and **Plug-in Config**. Then, click **Next**.

A template is provided in the field of **Plug-in Config**, which includes inputs and processors. You can configure the parameters based on your business requirements.

- inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

  > **Note** You can specify only one type of data source in inputs.

- processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

```
{
 "inputs": [
     {
         "type": "service_canal",
         "detail": {
             "Host": "************.mysql.rds.aliyuncs.com",
             "Port": 3306,
             "User" : "root",
             "ServerID" : 56321,
             "Password": "*******",
             "IncludeTables": [
                 "user_info\\..*"
             ],
             "ExcludeTables": [
                 ".*\\.\\S+_inner"
             ],
             "TextToString" : true,
             "EnableDDL" : true
         }
     }
 ]
}
```

Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_canal.
Host	string	No	The address of your database. If you do not configure this parameter, the default value 127.0.0.1 is used.
Port	int	No	The port of your database. If you do not configure this parameter, the default value 3306 is used.

Parameter	Type	Required	Description

User	string	No	The username of the account that you use to log on to your database. If you do not configure this parameter, the default value root is used.
			Make sure that the account is granted read permissions on the database and the REPLICATION permission. Example:
			`CREATE USER canal IDENTIFIED BY 'canal';`
			`GRANT SELECT, REPLICATION SLAVE, REPLICATION CLIENT ON *.* TO 'canal'@'%';`
			`-- GRANT ALL PRIVILEGES ON *.* TO 'canal'@'%' ;`
			`FLUSH PRIVILEGES;`

Password	string	No	The password of the account that you use to log on to your database. If you do not configure this parameter, an empty string is used. If you have high requirements for data security, we recommend that you set the username and password to xxx. After the settings are synchronized to the Logtail server, find and modify the parameters in the */usr/local/ilogtail/user_log_config.json* file. For more information, see Modify the Logtail configuration on the Logtail server. **Note** If you modify this parameter in the console, the parameter setting in the Logtail configuration on the Logtail server is overwritten after the modification is synchronized to the server.
ServerID	int	No	The ID of the slave MySQL node whose role Logtail assumes. If you do not configure this parameter, the default value 125 is used. **Note** The value of the ServerID parameter must be unique for your database. Otherwise, data collection fails.

Parameter	Type	Required	Description
IncludeTables	string array	Yes	The names of tables from which data is collected. A name must include the name of the database to which a table belongs and the name of the table. Example: test_db.test_table. You can specify a regular expression for the parameter. If a table does not meet any of the conditions specified by the IncludeTables parameter, the data in the table is not collected. If you want to collect data from all tables, you can set the value to .*\\..*. **Note** To implement exact match, add ^ to the start of the value and $ to the end. Example: ^test_db\\.test_table$.
ExcludeTables	string array	No	The names of tables from which data is not collected. A name must include the name of the database to which a table belongs and the name of the table. Example: test_db.test_table. You can specify a regular expression for the parameter. If a table meets one of the conditions specified by the ExcludeTables parameter, the data in the table is not collected. If you do not configure this parameter, the data in all tables is collected. **Note** To implement exact match, add ^ to the start of the value and $ to the end. Example: ^test_db\\.test_table$.

Parameter	Type	Required	Description

StartBinName	string	No	The name of the binary log file from which Logtail starts to collect data for the first time. If you do not configure this parameter, Logtail starts to collect data from the current time.
			If you want Logtail to collect data from a specified position, set StartBinName to the name of the binary log file from which you want Logtail to collect data and set StartBinlogPos to the offset of the file. Example:
			`# Set StartBinName to "mysql-bin.000063" and StartBinlogPos to 0.`
			`mysql> show binary logs;`
			`+------------------+-----------+`
			`| Log_name | File_size |`
			`+------------------+-----------+`
			`| mysql-bin.000063 | 241 |`
			`| mysql-bin.000064 | 241 |`
			`| mysql-bin.000065 | 241 |`
			`| mysql-bin.000066 | 10778 |`
			`+------------------+-----------+`
			`4 rows in set (0.02 sec)`
			Note
			If you specify a value for StartBinName, a large volume of traffic is generated when Logtail collects data for the first time.

StartBinlogPos	int	No	The offset of the binary log file from which Logtail starts to collect data for the first time. If you do not configure this parameter, the default value 0 is used.
EnableGTID	bool	No	Specifies whether to add GTIDs. If you do not configure this parameter, the default value true is used. If you set the value to false, no GTIDs are added to the data that is uploaded to Log Service.
EnableInsert	bool	No	Specifies whether to collect the data on INSERT events. If you do not configure this parameter, the default value true is used. If you set the value to false, Logtail does not collect the data on INSERT events.
EnableUpdate	bool	No	Specifies whether to collect the data on UPDATE events. If you do not configure this parameter, the default value true is used. If you set the value to false, Logtail does not collect the data on UPDATE events.

Parameter	Type	Required	Description
EnableDelete	bool	No	Specifies whether to collect the data on DELETE events. If you do not configure this parameter, the default value true is used. If you set the value to false, Logtail does not collect the data on DELETE events.
EnableDDL	bool	No	Specifies whether to collect the data on DDL events. If you do not configure this parameter, the default value false is used. This value indicates that Logtail does not collect the data on DDL events. **Note** If you set the value to true, the IncludeTables and ExcludeTables parameters become unavailable.
Charset	string	No	The encoding format. If you do not configure this parameter, the default value utf-8 is used.
TextToString	bool	No	Specifies whether to convert the data of the text type to the string type. If you do not configure this parameter, the default value false is used. This value indicates that the data type is not converted.
PackValues	bool	No	Specifies whether to pack event data in the JSON format. If you do not configure this parameter, the default value false is used. This value indicates that Logtail does not pack event data. If you set the value to true, Logtail packs event data into the data and old_data fields in the JSON format. The old_data field is available only for ROW_UPDATE events. For example, a table contains three columns named c1, c2, and c3. If you set the value to false, the data on ROW_INSERT events contains the c1, c2, and c3 fields. If you set the value to true, Logtail packs all data in the c1, c2, and c3 columns into the data field whose values are in the `{"c1":"...", "c2": "...", "c3": "..."}` format. **Note** This parameter is available only for Logtail V0.16.19 and later.

Parameter	Type	Required	Description
EnableEventMeta	bool	No	Specifies whether to collect the metadata of events. Default value: false. This value indicates that Logtail does not collect the metadata of events. The metadata of binary log events includes event_time, event_log_position, event_size, and event_server_id. **Note** This parameter is available only for Logtail V0.16.21 and later.

After the Logtail configuration is delivered to the Logtail server, Logtail immediately collects and sends data to Log Service when changes are made to your database.

Note By default, Logtail collects the incremental data of binary logs.

Modify the Logtail configuration on the Logtail server

If you did not enter real information for parameters such as Host, User, and Password in **Plug-in Config** when you created the Logtail configuration, you can modify the parameters after the Logtail configuration is delivered to the Logtail server.

1. Log on to the Logtail server.
2. Find the service_canal keyword in the */usr/local/ilogtail/user_log_config.json* file and modify parameters such as Host, User, and Password.
3. Run the following command to restart Logtail:

```
sudo /etc/init.d/ilogtaild stop; sudo /etc/init.d/ilogtaild start
```

What's next

After Logtail collects and sends MySQL binary logs to Log Service, you can view the logs in the Log Service console. For example, after you perform the `INSERT`, `UPDATE`, and `DELETE` operations on the `SpecialAlarm` table of the `user_info` database, Logtail collects and sends binary logs to Log Service. The following list describes the schema of the table, the operations that are performed on the table, and the collected logs.

- Table schema

```
CREATE TABLE `SpecialAlarm` (
`id` int(11) unsigned NOT NULL AUTO_INCREMENT,
`time` datetime NOT NULL,
`alarmtype` varchar(64) NOT NULL,
`ip` varchar(16) NOT NULL,
`count` int(11) unsigned NOT NULL,
PRIMARY KEY (`id`),
KEY `time` (`time`) USING BTREE,
KEY `alarmtype` (`alarmtype`) USING BTREE
) ENGINE=MyISAM AUTO_INCREMENT=1;
```

- Database operations

Perform the INSERT, DELETE, and UPDATE operations.

```
insert into specialalarm (`time`, `alarmType`, `ip`, `count`) values(now(), "NO_ALARM", "
10.10.**.***", 55);
delete from specialalarm where id = 4829235  ;
update specialalarm set ip = "10.11.***.**" where id = "4829234";
```

Create an index for `zc.specialalarm`.

```
ALTER TABLE `zc`.`specialalarm`
ADD INDEX `time_index` (`time` ASC);
```

- Collected logs

  You can view the logs that are collected for each operation on the search and analysis page of the Logstore that is specified in the Logtail configuration. Examples:

  - INSERT statement

    ```
    __source__:  10.30.**.**
    __tag__:__hostname__:  iZbp145dd9fccu*****
    __topic__:
    _db_:  zc
    _event_:  row_insert
    _gtid_:  7d2ea78d-b631-11e7-8afb-00163e0eef52:536
    _host_:  *********.mysql.rds.aliyuncs.com
    _id_:  113
    _table_:  specialalarm
    alarmtype:  NO_ALARM
    count:  55
    id:  4829235
    ip:  10.10.***.***
    time:  2017-11-01 12:31:41
    ```

  - DELETE statement

    ```
    __source__:  10.30.**.**
    __tag__:__hostname__:  iZbp145dd9fccu****
    __topic__:
    _db_:  zc
    _event_:  row_delete
    _gtid_:  7d2ea78d-b631-11e7-8afb-00163e0eef52:537
    _host_:  *********.mysql.rds.aliyuncs.com
    _id_:  114
    _table_:  specialalarm
    alarmtype:  NO_ALARM
    count:  55
    id:  4829235
    ip:  10.10.**.***
    time:  2017-11-01 12:31:41
    ```

- UPDATE statement

```
__source__:  10.30.**.**
__tag__:__hostname__:  iZbp145dd9fccu****
__topic__:
_db_:  zc
_event_:  row_update
_gtid_:  7d2ea78d-b631-11e7-8afb-00163e0eef52:538
_host_:  *********.mysql.rds.aliyuncs.com
_id_:  115
_old_alarmtype:  NO_ALARM
_old_count:  55
_old_id:  4829234
_old_ip:  10.10.22.133
_old_time:  2017-10-31 12:04:54
_table_:  specialalarm
alarmtype:  NO_ALARM
count:  55
id:  4829234
ip:  10.11.***.***
time:  2017-10-31 12:04:54
```

- DDL statement

```
__source__:  10.30.**.**
__tag__:__hostname__:  iZbp145dd9fccu****
__topic__:
_db_:  zc
_event_:  row_update
_gtid_:  7d2ea78d-b631-11e7-8afb-00163e0eef52:539
_host_:  *********.mysql.rds.aliyuncs.com
ErrorCode:  0
ExecutionTime:  0
Query:  ALTER TABLE `zc`.`specialalarm`
ADD INDEX `time_index` (`time` ASC)
StatusVars:
```

Field	Description
`_host_`	The hostname of the database.
`_db_`	The name of the database.
`_table_`	The name of the table.
`_event_`	The type of the event.
`_id_`	The auto-increment ID. IDs start from 0 and increment by 1 each time the data on a binary log event is collected.
`_gtid_`	The GTID.
`_filename_`	The name of the binary log file.

Field	Description
`_offset_`	The offset of the binary log file. The value is updated only when a COMMIT operation is performed.

3.7.3. Collect MySQL query results

This topic describes how to create a Logtail configuration in the Log Service console to collect MySQL query results.

Prerequisites

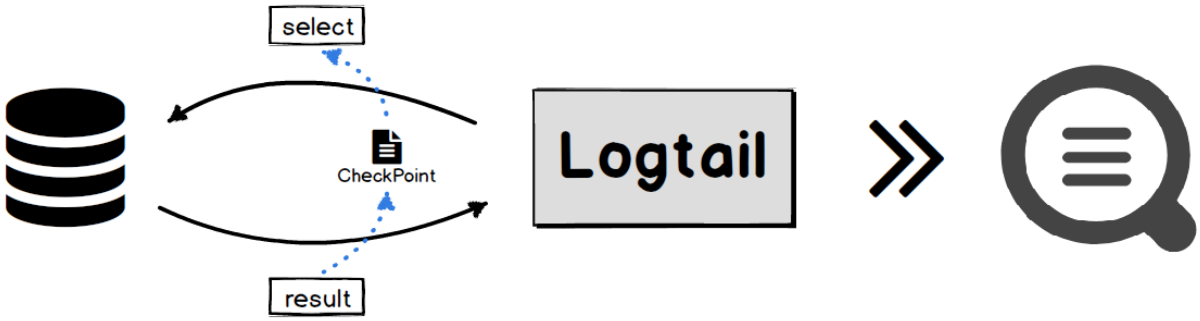
Logtail is installed on a server. For ease of understanding, this server is referred to as the Logtail server in this topic. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

> **Note** Linux servers support Logtail V0.16.0 or later. Windows servers support Logtail V1.0.0.8 or later.

Principle

Logtail executes the SELECT statement that is specified in a Logtail configuration on a regular basis, and then uploads the query results to Log Service.

After Logtail obtains query results, Logtail saves the value of the CheckPoint field in the results to the Logtail server. The next time Logtail executes the SELECT statement, Logtail adds the value of the CheckPoint field to the SELECT statement. This way, Logtail can collect incremental data.



Features

- MySQL databases are supported.
- You can configure paged query settings.
- You can set time zones.
- You can specify timeout periods.
- The values of the CheckPoint field can be saved.
- SSL is supported.
- You can specify the maximum size of data that can be collected at a time.

Scenarios

- Collect incremental data based on marks such as an auto-increment ID or a point in time.
- Customize data synchronization based on filter conditions.

Procedure

1.
2. In the **Import Data** section, click **MySQL Query Result - Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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, configure **Config Name** and **Plug-in Config**. Then, click **Next**.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

```
{
  "inputs": [
    {
      "type": "service_mysql",
      "detail": {
        "Address": "************.mysql.rds.aliyuncs.com",
        "User": "****",
        "Password": "*******",
        "DataBase": "****",
        "Limit": true,
        "PageSize": 100,
        "StateMent": "select * from db.VersionOs where time > ?",
        "CheckPoint": true,
        "CheckPointColumn": "time",
        "CheckPointStart": "2018-01-01 00:00:00",
        "CheckPointSavePerPage": true,
        "CheckPointColumnType": "time",
        "IntervalMs": 60000
      }
    }
  ]
}
```

Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_mysql.
Address	string	No	The address of your MySQL database. If you do not configure this parameter, the default value 127.0.0.1:3306 is used.
User	string	No	The username of the account that you use to log on to your MySQL database. If you do not configure this parameter, the default value root is used.

Parameter	Type	Required	Description
Password	string	No	The password of the account that you use to log on to your MySQL database. If you do not configure this parameter, an empty string is used. If you have high requirements for data security, we recommend that you set the username and password to xxx. After the settings are synchronized to the Logtail server, find and modify the parameters in the */usr/local/ilogtail/user_log_config.json* file. For more information, see Modify the Logtail configuration on the Logtail server. **Note** If you modify this parameter in the console, the parameter setting in the Logtail configuration on the Logtail server is overwritten after the modification is synchronized to the server.
DataBase	string	No	The name of your MySQL database.
DialTimeOutMs	int	No	The timeout period for connections to your MySQL database. Unit: milliseconds. If you do not configure this parameter, the default value 5000 is used.
ReadTimeOutMs	int	No	The timeout period for data read from your MySQL database. Unit: milliseconds. If you do not configure this parameter, the default value 5000 is used.
StateMent	string	No	The SQL statement. If you set the CheckPoint parameter to true, you must include the CheckPointColumn parameter in a WHERE clause of the SQL statement that you specify for the StateMent parameter. You must also set the value of the column specified by the CheckPointColumn parameter to ?. For example, if you set the CheckPointColumn parameter to id, you must set the value of the StateMent parameter in the `SELECT * from ... where id > ?` format.

Parameter	Type	Required	Description
Limit	boolean	No	Specifies whether to use a LIMIT clause to perform paged queries. If you do not configure this parameter, the default value false is used. This value indicates that no LIMIT clause is used. We recommend that you use a LIMIT clause to perform paged queries. If you set the Limit parameter to true, a LIMIT clause is automatically added to the SQL statement that you specify for the StateMent parameter when Logtail executes the SQL statement.
PageSize	int	No	The maximum number of logs that can be returned on each page. If you set the Limit parameter to true, you must configure this parameter.
MaxSyncSize	int	No	The maximum number of logs that can be synchronized at a time. If you do not configure this parameter, the default value 0 is used. This value indicates that the number is not limited.
CheckPoint	boolean	No	Specifies whether to use checkpoints to collect data. If you do not configure this parameter, the default value false is used. This value indicates that no checkpoints are used.
CheckPointColumn	string	No	The name of the checkpoint column. If you set the CheckPoint parameter to true, you must configure this parameter. **Notice** Values in the checkpoint column must be incremental. Otherwise, some data may not be collected. The maximum value in the results of a query operation is used as the input for the next query operation.
CheckPointColumnType	string	No	The type of the checkpoint column. Valid values: int and time. If you set this parameter to int, the values in the checkpoint column are of the int64 type. If you set this parameter to time, the values in the checkpoint column can be of the date, datetime, or time type that is supported by MySQL. If you set the CheckPoint parameter to true, you must configure this parameter.

Parameter	Type	Required	Description
CheckPointStart	string	No	The initial value of checkpoints. If you set the CheckPoint parameter to true, you must configure this parameter.
CheckPointSavePerPage	boolean	No	If you set this parameter to true, a checkpoint is recorded for each page. If you set this parameter to false, a checkpoint is recorded each time data is synchronized to Log Service.
IntervalMs	int	Yes	The interval of data synchronization. Unit: milliseconds.

Modify the Logtail configuration on the Logtail server

If you did not enter real information for parameters such as Address, User, and Password in **Plug-in Config** when you created the Logtail configuration, you can modify the parameters after the Logtail configuration is delivered to the Logtail server.

1. Log on to the Logtail server.
2. Find the service_mysql keyword in the */usr/local/ilogtail/user_log_config.json* file and modify parameters such as Address, User, and Password.
3. Run the following command to restart Logtail:

```
sudo /etc/init.d/ilogtaild stop; sudo /etc/init.d/ilogtaild start
```

What's next

After Logtail collects and sends MySQL query results to Log Service, you can view the results in the Log Service console. The following list describes a table schema and a sample of log data that is collected.

- Table schema

```
CREATE TABLE `VersionOs` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT COMMENT 'id',
  `time` datetime NOT NULL,
  `version` varchar(10) NOT NULL DEFAULT '',
  `os` varchar(10) NOT NULL,
  `count` int(11) unsigned NOT NULL,
  PRIMARY KEY (`id`),
  KEY `timeindex` (`time`)
)
```

- Sample log

```
"count":  "4"
"id:  "721097"
"os:  "Windows"
"time:  "2017-08-25 13:00:00"
"version":  "1.3.0"
```

3.7.4. Collect HTTP data

You can create a Logtail configuration file in the Log Service to collect HTTP data from specified URLs. After the Logtail configuration file is synchronized to the server on which Logtail is installed, Logtail sends requests at a regular interval to the specified URLs. Then, Logtail uploads the content of the response body as a data source to Log Service. This topic describes how to configure Logtail in the Log Service console to collect HTTP data.

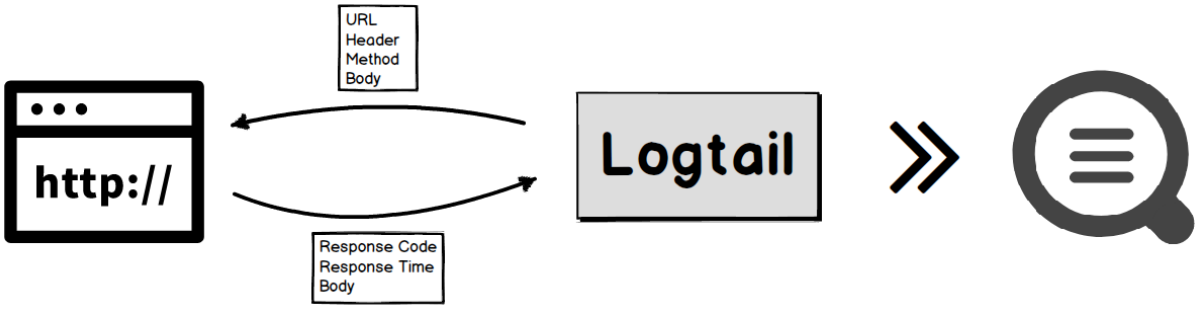
Prerequisites

Logtail is installed on the server that you use to collect HTTP data. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

> **Note** Servers that run Linux support Logtail 0.16.0 or later. Servers that run Windows support Logtail 1.0.0.8 or later.

Implementation

Logtail initiates regular HTTP requests based on the URLs, methods, headers, and bodies specified in the Logtail configurations. After Logtail receives a response, Logtail uploads the response status code, the content of the response body, and the response time to Log Service.



Features

- Supports multiple URLs.
- Allows you to set multiple HTTP methods.
- Allows you to set the interval at which HTTP requests are initiated.
- Allows you to customize request headers.
- Supports HTTPS.
- Allows you to check whether the content of the request body matches a fixed pattern.

Scenarios

- Monitor application status by using HTTP APIs.
  - NGINX
  - Docker
  - Elasticsearch
  - HAProxy
  - Other services that provide monitoring HTTP APIs

- Monitor service availability.

  Logtail monitors the availability of a service by sending requests at a regular interval to the service and checking the response status code and latency.

- Retrieve data such as tweets and the number of followers at a regular interval.

Limits

- A URL must start with `http` or `https`.
- Custom certificates are not supported.
- Interactive communications are not supported.

Procedure

The following procedure shows how to collect data about the NGINX status module. Requests are sent to the URL `http://127.0.0.1/ngx_status` every 1,000 milliseconds. A regular expression is used to extract the status information from the response body.

1.
2. In the **Import Data** section, select **Custom Data Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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, set the **Config Name** and **Plug-in Config** parameters.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

> **Note** You can specify only one type of data source in inputs.

- processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

```
{
 "inputs": [
     {
         "type": "metric_http",
         "detail": {
             "IntervalMs": 1000,
             "Addresses": [
                 "http://127.0.0.1/ngx_status"
             ],
             "IncludeBody": true
         }
     }
 ],
 "processors" : [
     {
         "type": "processor_regex",
         "detail" : {
             "SourceKey": "content",
             "Regex": "Active connections: (\\d+)\\s+server accepts handled requests\\s
+(\\d+)\\s+(\\d+)\\s+(\\d+)\\s+Reading: (\\d+) Writing: (\\d+) Waiting: (\\d+). *",
             "Keys": [
                 "connection",
                 "accepts",
                 "handled",
                 "requests",
                 "reading",
                 "writing",
                 "waiting"
             ],
             "FullMatch": true,
             "NoKeyError": true,
             "NoMatchError": true,
             "KeepSource": false
         }
     }
 ]
}
```

Parameter	Type	Required	Description
type	String.	Yes	The type of the data source. Set the value to metric_http.

Parameter	Type	Required	Description
Addresses	String. array.	Yes	The URLs to which requests are sent. **Note** A URL must start with `http` or `https`.
IntervalMs	Int.	Yes	The interval between two successive requests. Unit: milliseconds.
Method	String.	No	The request method. Default value: `GET`. The value must be uppercase letters.
Body	String.	No	The content of the HTTP request body. Default value: null.
Headers	Key: string. Value: string map.	No	The content of the HTTP request header. Default value: null.
PerAddressSleepMs	Int.	No	The interval at which requests are sent to URLs that are specified by the Addresses parameter. Unit: milliseconds. Default value: 100.
ResponseTimeoutMs	Int.	No	The timeout period for a request. Unit: milliseconds. Default value: 5000.
IncludeBody	Boolean.	No	Specifies whether to collect the request body. Default value: false. If you set the value to true, the content of the request body is stored in the field named content.
FollowRedirects	Boolean.	No	Specifies whether to automatically process URL redirects. Default value: false.
InsecureSkipVerify	Boolean.	No	Specifies whether to skip the HTTPS security check. Default value: false.
ResponseStringMatch	String.	No	Specifies whether to match the response body by using a regular expression. The result is saved to the field named _response_match_. If the response body matches the regular expression, the value of the field is yes. Otherwise, the value is no.

Result

After data is collected, you can view the data in the Log Service console. In addition to the data that is parsed by using the regular expression, you can view the HTTP request method, request URL, response time, status code, and request result.

```
"Index" : "7"
"connection" : "1"
"accepts" : "6079"
"handled" : "6079"
"requests" : "11596"
"reading" : "0"
"writing" : "1"
"waiting" : "0"
"_method_" : "GET"
"_address_" : "http://127.0.0.1/ngx_status"
"_response_time_ms_" : "1.320"
"_http_response_code_" : "200"
"_result_" : "success"
```

By default, the following fields are uploaded for each request.

Field	Description
address	The request URL.
method	The request method.
_response_time_ms_	The response latency. Unit: milliseconds.
_http_response_code_	The HTTP status code.
result	The request result. Valid values: success, invalid_body, match_regex_invalid, mismatch, and timeout.
_response_match_	Specifies whether the content of the response body matches the regular expression that is specified by the ResponseStringMatch parameter. If the `ResponseStringMatch` parameter is not specified, the value of this field is null. If the ResponseStringMatch parameter is specified, the value is yes or no.

3.7.5. Collect syslogs

This topic describes how to create a Logtail configuration in the Log Service console to collect syslogs.

Prerequisites

Logtail is installed on a server. For ease of understanding, this server is referred to as the Logtail server in this topic. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

> **Note** Linux servers support Logtail V0.16.13 or later. Windows servers support Logtail V1.0.0.8 or later.

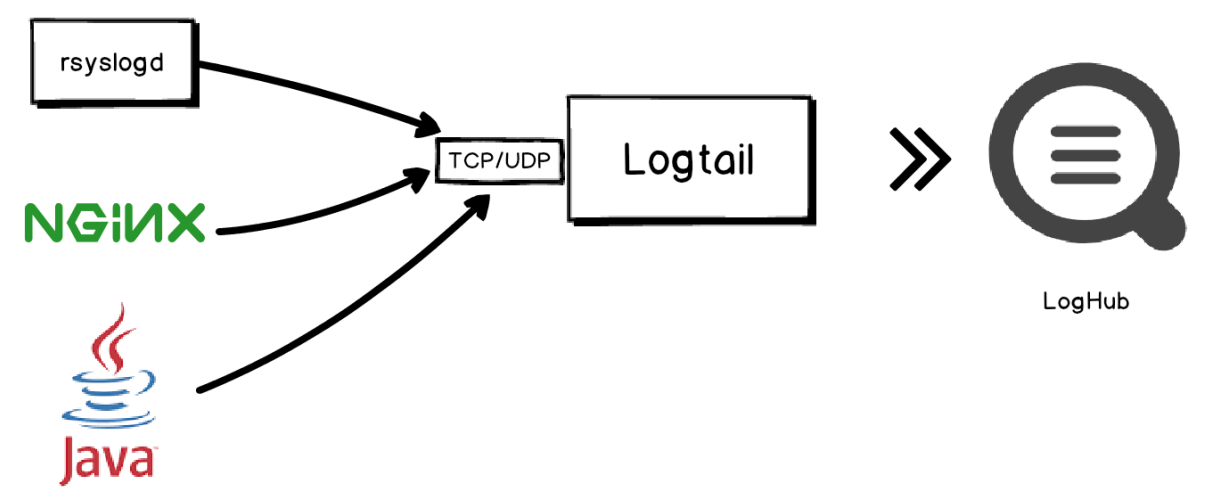
Overview

Linux servers allow you to use syslog agents such as rsyslog to forward local syslogs to the IP address and port of a specified server. After you apply a Logtail configuration to the specified server, the Logtail plug-in specified in the configuration receives the forwarded syslogs over TCP or UDP. The plug-in also parses the syslogs based on the specified syslog protocol, and extracts the facility, tag(program), severity, and content fields from the syslogs. The syslog protocols defined in RFC 3164 and RFC 5424 are supported.

You can configure multiple Logtail plug-ins based on your business requirements. For example, you can configure two Logtail plug-ins to listen on 127.0.0.1:9000 over both TCP and UDP.

Principle

After you configure Logtail plug-ins to listen on a specified address and port, Logtail collects and sends data to Log Service. The data includes the system logs that are collected by using rsyslog, the access logs or error logs that are forwarded by NGINX, and the logs that are forwarded by syslog clients.



Configure Logtail plug-ins to collect syslogs

1. Add a forwarding rule for rsyslog.
   i. Modify the */etc/rsyslog.conf* configuration file of rsyslog on the server from which you want to collect syslogs. (For ease of understanding, this server is referred to as the syslog server in this section.) Add a forwarding rule to the end of the configuration file.

      After the forwarding rule is added, rsyslog forwards syslogs to a specified IP address and port.

      - If Logtail resides on the syslog server, you must specify the IP address 127.0.0.1 and a non-well-known port that is unoccupied in the forwarding rule.
      - If Logtail resides on a different server from the syslog server, you must specify the public IP address of the different server and a non-well-known port that is unoccupied in the forwarding rule.

      The following example shows a forwarding rule, which allows all syslogs to be forwarded to 127.0.0.1:9000 over TCP. For more information about the configuration file, see RSyslog Documentation.

      ```
      *.* @@127.0.0.1:9000
      ```

ii. Run the following command to restart rsyslog and validate the forwarding rule:

```
sudo service rsyslog restart
```

2.

3. In the **Import Data** section, click **Custom Data Plug-in**.
4. Select the project and Logstore. Then, click **Next**.
5. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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.
6. Select the new machine group from **Source Server Groups** and move the machine group to **Applied Server Groups**. Then, click **Next**.

   > **Notice** If you apply a machine group immediately after you create the machine group, the heartbeat status of the machine group may be **FAIL**. This issue occurs because the machine group is not connected to Log Service. To resolve this issue, you can click **Automatic Retry**. If the issue persists, see What do I do if no heartbeat connections are detected on Logtail?

7. In the **Specify Data Source** step, configure **Config Name** and **Plug-in Config**. Then, click **Next**.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

   The following example shows how to configure Logtail plug-ins to listen on 127.0.0.1 over both UDP and TCP:

```
{
    "inputs": [
        {
            "type": "service_syslog",
            "detail": {
                "Address": "tcp://127.0.0.1:9000",
                "ParseProtocol": "rfc3164"
            }
        },
        {
            "type": "service_syslog",
            "detail": {
                "Address": "udp://127.0.0.1:9001",
                "ParseProtocol": "rfc3164"
            }
        }
    ]
}
```

Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_syslog.
Address	string	No	The listening protocol, address, and port that are used by a Logtail plug-in. The plug-in listens on and obtains data based on the Logtail configuration. Specify the value in the [tcp/udp]://[*ip*]:[*port*] format. If you do not configure this parameter, the default value tcp://127.0.0.1:9999 is used. **Note** ◦ The listening protocol, address, and port that you specify must be the same as those specified in the forwarding rule that is added to the configuration file of rsyslog. ◦ If the Logtail server uses multiple IP addresses to receive data, set the IP address to 0.0.0.0. This address indicates that the plug-in listens on all the IP addresses of the server.

Parameter	Type	Required	Description
ParseProtocol	string	No	The protocol that is used to parse syslogs. By default, this parameter is empty, which indicates that syslogs are not parsed. Valid values: ○ rfc3164: The RFC 3164 protocol is used to parse syslogs. ○ rfc5424: The RFC 5424 protocol is used to parse syslogs. ○ auto: The plug-in automatically selects a protocol based on the content of syslogs.
IgnoreParseFailure	boolean	No	The operation that is performed after a syslog fails to be parsed. If you do not configure this parameter, the default value true is used. This value indicates that a syslog that fails to be parsed is directly included in the content field that is returned. If you set the value to false, a syslog is discarded after it fails to be parsed.

Configure Logtail plug-ins to collect NGINX logs

NGINX servers allow you to directly forward access logs to specified IP addresses and ports by using the syslog protocol. If you want to deliver all the data of a server as syslogs to Log Service, you can create a Logtail configuration to collect the data. The data includes NGINX access logs.

1. Add a forwarding rule for NGINX.
   i. Find the *nginx.conf* file on the NGINX server and add a forward rule to the end of the file. For more information, see NGINX Beginner's Guide.

      The following sample code provides an example of a forwarding rule:

      ```
      http {
          ...
          # Add this line.
          access_log syslog:server=127.0.0.1:9000,facility=local7,tag=nginx,severity=info
      combined;
          ...
      }
      ```

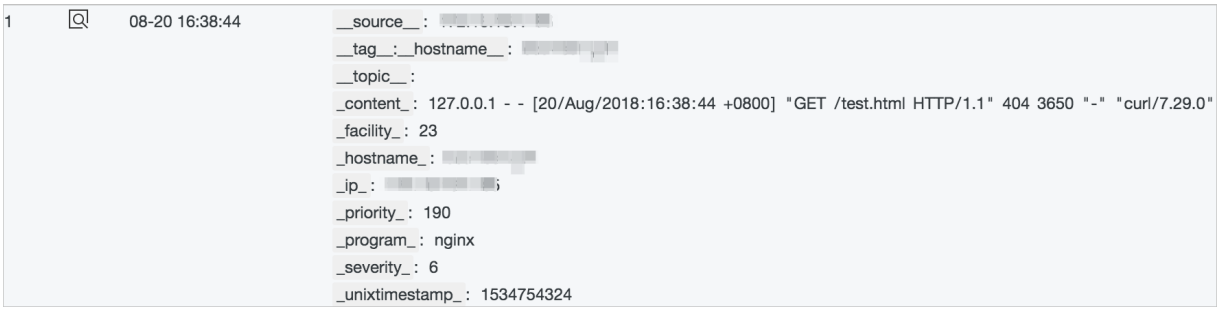
   ii. Run the following command to restart the NGINX service and validate the forwarding rule:

      ```
      sudo service nginx restart
      ```

2. Create a Logtail configuration. For more information, see Configure Logtail plug-ins to collect syslogs.

What's next

After Logtail collects and sends syslogs to Log Service, you can view the logs in the Log Service console.



Field	Description
hostname	The hostname. If no hostname is included in the log, the hostname of the current host is obtained.
program	The tag field in the syslog protocol.
priority	The priority field in the syslog protocol.
facility	The facility field in the syslog protocol.
severity	The severity field in the syslog protocol.
unixtimestamp	The timestamp of the log.
content	The content of the log. If the log fails to be parsed, this field contains the content of the log that is not parsed.
_ip_	The IP address of the current host.
_client_ip_	The IP address of Logtail that transfers the log.

3.7.6. Collect data from Beats and Logstash

This topic describes how to configure Logtail in the Log Service console to collect data from Beats and Logstash.

Prerequisites

- Logtail is installed on the server that you use to collect data from Beats and Logstash. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

  > **Note** Servers that run Linux support Logtail 0.16.9 or later. Servers that run Windows support Logtail 1.0.0.8 or later.

- Data is collected by using Logstash or Beats.
  - For more information about how to collect data from Logstash, visit Logstash-Lumberjack-Output.

- For more information about how to collect data from Beats, visit Beats-Lumberjack-Output.

  The procedure in this topic describes how to use Packetbeat to collect data transmitted on the local network, and use the Logtail Lumberjack plug-in to upload the data to Log Service. Data collected by using Packetbeat is sent to Logstash, as shown in the following sample script:

  ```
  output.logstash:
    hosts: ["127.0.0.1:5044"]
  ```

Context

Logstash and Beats (such as MetricBeat, PacketBeat, Winlogbeat, Auditbeat, Filebeat, and Heartbeat) support the Lumberjack protocol. Therefore, Logtail can use the protocol to upload data that is collected by Beats and Logstash to Log Service.

> **Note**
> - You can configure multiple Lumberjack plug-ins, but these plug-ins cannot listen on the same port.
> - Lumberjack plug-ins support SSL. Data uploaded to Log Service from Logstash must be encrypted by using SSL.

Procedure

1.
2. In the **Import Data** section, select **Custom Data Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     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, set the **Config Name** and **Plug-in Config** parameters.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

   Data from Beats and Logstash is in the JSON format. `processor_anchor` is configured to expand the JSON-formatted data.

```
{
  "inputs": [
    {
      "detail": {
        "BindAddress": "0.0.0.0:5044"
      },
      "type": "service_lumberjack"
    }
  ],
  "processors": [
    {
      "detail": {
        "Anchors": [
          {
            "ExpondJson": true,
            "FieldType": "json",
            "Start": "",
            "Stop": ""
          }
        ],
        "SourceKey": "content"
      },
      "type": "processor_anchor"
    }
  ]
}
```

Parameter	Type	Required	Description
type	String	Yes	The type of the data source. Set the value to service_lumberjack.
BindAddress	String	No	The IP address and port of the server to which data can be sent by using the Lumberjack protocol. Default value: 127.0.0.1:5044. To enable access from other hosts in the LAN by using the Lumberjack protocol, set the value to 0.0.0.0:5044.

Parameter	Type	Required	Description
V1	Boolean	No	Specifies whether to use the Lumberjack protocol v1. Default value: false. Logstash supports the Lumberjack protocol v1.
V2	Boolean	No	Specifies whether to use the Lumberjack protocol v2. Default value: true. Beats support the Lumberjack protocol v2.
SSLCA	String	No	The path of the Certificate Authority that issues the signature certificate. Default value: null. If you use a self-signed certificate, you do not need to specify the parameter.
SSLCert	String	No	The path of the certificate. Default value: null.
SSLKey	String	No	The path of the private key that corresponds to the certificate. Default value: null.
InsecureSkipVerify	Boolean	No	Specifies whether to skip the SSL security check. Default value: false. This value indicates the SSL security check is performed.

What's next

After Logtail uploads data to Log Service, you can view the data in the Log Service console. The following content is the sample data uploaded to Log Service.

```
_@metadata_beat:  packetbeat
_@metadata_type:  doc
_@metadata_version:  6.2.4
_@timestamp:  2018-06-05T03:58:42.470Z
__source__:  **. **. **.**
__tag__:__hostname__:  *******
__topic__:
_beat_hostname:  bdbe0b8d53a4
_beat_name:  bdbe0b8d53a4
_beat_version:  6.2.4
_bytes_in:  56
_bytes_out:  56
_client_ip:  192.168.5.2
_icmp_request_code:  0
_icmp_request_message:  EchoRequest(0)
_icmp_request_type:  8
_icmp_response_code:  0
_icmp_response_message:  EchoReply(0)
_icmp_response_type:  0
_icmp_version:  4
_ip:  127.0.0.1
_path:  127.0.0.1
_responsetime:  0
_status:  OK
_type:  icmp
```

3.7.7. Collect systemd-journald logs

You can use Logtail to collect Linux systemd-journald logs from binary files. This topic describes how to configure Logtail in the Log Service console to collect systemd-journald logs.

Prerequisites

Logtail is installed on the server that you use to collect systemd-journald logs. For more information, see Install Logtail on a Linux server.

> **Note** Only Linux servers that run Logtail 0.16.18 or later are supported.

Overview

systemd is a system and service manager for Linux. When systemd runs as an initialization process (PID = 1), it boots and maintains the services of user spaces. systemd manages kernel and application logs in a centralized manner. The configuration file that is used to manage these logs is */etc/systemd/journald.conf*.

> **Note** The operating system on which systemd runs must support the journal log format.

Features

- Allows you to set the initial collection position. Checkpoints are automatically saved for subsequent

data collection. The process is not affected when applications are restarted.

- Allows you to filter specified applications.
- Allows you to collect kernel logs.
- Supports automatic parsing of log severity.
- Allows you to run systemd as a container to collect journal logs from hosts. This feature is applicable when you collect logs from Docker and Kubernetes clusters.

Scenarios

- Monitor kernel events, and trigger alerts when exceptions occur.
- Collect system logs for long-term storage and release disk space.
- Collect application logs for analysis or alerting.
- Collect all journal logs and query log data with higher efficiency than journalctl.

Procedure

1.
2. In the **Import Data** section, select **Custom Data Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, 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. ECS instances are used as an example.
     a. Install Logtail on ECS instances. For more information, see Install Logtail on ECS instances.

     If Logtail is installed on the ECS instances, click **Complete Installation**.

     > **Note** If you want to collect logs from self-managed clusters or servers of third-party cloud service providers, you must install Logtail on these servers. For more information, see Install Logtail on a Linux server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     b. After you install Logtail, click **Complete Installation**.
     c. On the page that appears, set related parameters for the machine group. For more information, see Create an IP address-based machine group or Create a custom ID-based machine group.
5.
6. In the **Specify Data Source** step, set the **Config Name** and **Plug-in Config** parameters.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

```
{
  "inputs": [
    {
      "detail": {
        "JournalPaths": [
          "/var/log/journal"
        ],
        "Kernel": true,
        "ParsePriority": true,
        "ParseSyslogFacility": true
      },
      "type": "service_journal"
    }
  ]
}
```

Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_journal.
JournalPaths	String array	Yes	The path of journal logs. We recommend that you set the value to the directory of journal logs, for example, */var/log/journal*.
SeekPosition	string	No	The method that is used to collect logs for the first time. Valid values: head and tail. Default value: tail. ◦ If you set the value to head, all data is collected. ◦ If you set the value to tail, only the data that is generated after the Logtail configuration file takes effect is collected.
Kernel	bool	No	Specifies whether to collect kernel logs. Default value: true. This value indicates that kernel logs are collected.
Units	String array	No	The applications from which logs are collected. Default value: null. This value indicates that logs from all applications are collected.
ParseSyslogFacility	bool	No	Specifies whether to parse the facility field of syslog logs. Default value: false. This value indicates that the facility field of syslog logs is not parsed.

Parameter	Type	Required	Description

ParsePriority	bool	No	Specifies whether to parse the priority field of syslog logs. Default value: false. This value indicates that the priority field of syslog logs is not parsed.
			If you set the value to true, the priority field of syslog logs is parsed based on the following mapping relationships:
			`"0": "emergency"`
			`"1": "alert"`
			`"2": "critical"`
			`"3": "error"`
			`"4": "warning"`
			`"5": "notice"`
			`"6": "informational"`
			`"7": "debug"`

UseJournalEventTime	bool	No	Specifies whether to use the field of journal logs as the timestamp of the log. Default value: false. This value indicates that the time when a log entry is collected is recorded as the timestamp of the log entry. In most cases, the difference between the log collection time and the log timestamp is less than 3 seconds.

Examples

- Example 1

  To collect journal logs from the `/var/log/journal` directory, you can use the following configuration file:

```
{
  "inputs": [
    {
      "detail": {
        "JournalPaths": [
          "/var/log/journal"
        ]
      },
      "type": "service_journal"
    }
  ]
}
```

Sample log entry:

```
MESSAGE: rejected connection from "192.168.0.250:43936" (error "EOF", ServerName "")
PACKAGE: embed
PRIORITY: 6
SYSLOG_IDENTIFIER: etcd
_BOOT_ID: fe919cd1268f4721bd87b5c18afe59c3
_CAP_EFFECTIVE: 0
_CMDLINE: /usr/bin/etcd --election-timeout=3000 --heartbeat-interval=500 --snapshot-coun
t=50000 --data-dir=data.etcd --name 192.168.0.251-name-3 --client-cert-auth --trusted-ca-
file=/var/lib/etcd/cert/ca.pem --cert-file=/var/lib/etcd/cert/etcd-server.pem --key-file=
/var/lib/etcd/cert/etcd-server-key.pem --peer-client-cert-auth --peer-trusted-ca-file=/va
r/lib/etcd/cert/peer-ca.pem --peer-cert-file=/var/lib/etcd/cert/192.168.0.251-name-3.pem
--peer-key-file=/var/lib/etcd/cert/192.168.0.251-name-3-key.pem --initial-advertise-peer-
urls https://192.168.0.251:2380 --listen-peer-urls https://192.168.0.251:2380 --advertise
-client-urls https://192.168.0.251:2379 --listen-client-urls https://192.168.0.251:2379 -
-initial-cluster 192.168.0.249-name-1=https://192.168.0.249:2380,192.168.0.250-name-2=htt
ps://192.168.0.250:2380,192.168.0.251-name-3=https://192.168.0.251:2380 --initial-cluster
-state new --initial-cluster-token abac64c8-baab-4ae6-8412-4253d3cfb0cf
_COMM: etcd
_EXE: /opt/etcd-v3.3.8/etcd
_GID: 995
_HOSTNAME: iZbp1f7y2ikfe4l8nx95amZ
_MACHINE_ID: f0f31005fb5a436d88e3c6cbf54e25aa
_PID: 10926
_SOURCE_REALTIME_TIMESTAMP: 1546854068863857
_SYSTEMD_CGROUP: /system.slice/etcd.service
_SYSTEMD_SLICE: system.slice
_SYSTEMD_UNIT: etcd.service
_TRANSPORT: journal
_UID: 997
__source__: 172.16.1.4
__tag__:__hostname__: logtail-ds-8kqb9
__topic__:
_monotonic_timestamp_: 1467135144311
_realtime_timestamp_: 1546854068864309
```

- Example 2

  To collect important fields from system logs of Kubernetes hosts in DaemonSet mode, you can use the following configuration file:

```
{
  "inputs": [
    {
      "detail": {
        "JournalPaths": [
          "/logtail_host/var/log/journal"
        ],
        "ParsePriority": true,
        "ParseSyslogFacility": true
      },
      "type": "service_journal"
    }
  ],
  "processors": [
    {
      "detail": {
        "Exclude": {
          "UNIT": "^libcontainer.*test"
        }
      },
      "type": "processor_filter_regex"
    },
    {
      "detail": {
        "Include": [
          "MESSAGE",
          "PRIORITY",
          "_EXE",
          "_PID",
          "_SYSTEMD_UNIT",
          "_realtime_timestamp_",
          "_HOSTNAME",
          "UNIT",
          "SYSLOG_FACILITY",
          "SYSLOG_IDENTIFIER"
        ]
      },
      "type": "processor_pick_key"
    }
  ]
}
```

Sample log entry:

```
MESSAGE:  rejected connection from "192.168.0.251:48914" (error "EOF", ServerName "")
PRIORITY:  informational
SYSLOG_IDENTIFIER:  etcd
_EXE:  /opt/etcd-v3.3.8/etcd
_HOSTNAME:  iZbp1i0czq3zgvxlx7u8ueZ
_PID:  10590
_SYSTEMD_UNIT:  etcd.service
__source__:  172.16.0.141
__tag__:__hostname__:  logtail-ds-dp48x
__topic__:
_realtime_timestamp_:  1547975837008708
```

3.7.8. Collect Docker events

Docker events include all interactive events of objects such as containers, images, plug-ins, networks, and volumes. This topic describes how to configure Logtail in the Log Service console to collect Docker events.

Prerequisites

Logtail is installed on the server that you use to collect Docker events. For more information, see Install Logtail on a Linux server.

> **Note** Only Linux servers that run Logtail 0.16.18 or later are supported.

Limits

- Logtail that runs on containers or hosts must be authorized to access the `/var/run/docker.sock` file.

  For information about how to use Logtail to collect Kubernetes logs, see Collect Kubernetes logs. For information about how to collect standard container logs, see Collect logs from standard Docker containers.

- When Logtail is restarted or stopped, container events are not collected.

Scenarios

- Monitor the start and stop events of all containers, and trigger alerts when core containers stop running.
- Collect all container events for auditing, security analysis, and troubleshooting.
- Monitor all image pulling events, and trigger an alert if an image is pulled from an invalid path.

Procedure

1.
2. In the **Import Data** section, select **Custom Data Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, 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, ECS instances are used.

a. Install Logtail on ECS instances. For more information, see Install Logtail on ECS instances.

If Logtail is installed on the ECS instances, click **Complete Installation**.

> **Note** If you want to collect logs from self-managed clusters or servers of third-party cloud service providers, you must install Logtail on these servers. For more information, see Install Logtail on a Linux server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

b. After you install Logtail, click **Complete Installation**.

c. On the page that appears, set related parameters for the machine group. For more information, see Create an IP address-based machine group or Create a custom ID-based machine group.

5.

6. In the **Specify Data Source** step, set the **Config Name** and **Plug-in Config** parameters.

   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

```
{
  "inputs": [
    {
      "detail": {},
      "type": "service_docker_event"
    }
  ]
}
```

Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_docker_event.
EventQueueSize	int	No	The maximum number of events in the event queue. Default value: 10.

What's next

After Logtail collects Docker events and uploads the events to Log Service, you can view the events in the Log Service console. The following examples show multiple event log entries.

- Example 1: image pulling event

```
__source__:  10.10.10.10
__tag__:__hostname__:  logtail-ds-77brr
__topic__:
_action_:  pull
_id_:  registry.cn-hangzhou.aliyuncs.com/ringtail/eventer:v1.6.1.3
_time_nano_:  1547910184047414271
_type_:  image
name:  registry.cn-hangzhou.aliyuncs.com/ringtail/eventer
```

- Example 2: container destruction event in Kubernetes

```
__source__:  10.10.10.10
__tag__:__hostname__:  logtail-ds-xnvz2
__topic__:
_action_:  destroy
_id_:  af61340b0ac19e6f5f32be672d81a33fc4d3d247bf7dbd4d3b2c030b8bec4a03
_time_nano_:  1547968139380572119
_type_:  container
annotation.kubernetes.io/config.seen:  2019-01-20T15:03:03.114145184+08:00
annotation.kubernetes.io/config.source:  api
annotation.scheduler.alpha.kubernetes.io/critical-pod:
controller-revision-hash:  2630731929
image:  registry-vpc.cn-hangzhou.aliyuncs.com/acs/pause-amd64:3.0
io.kubernetes.container.name:  POD
io.kubernetes.docker.type:  podsandbox
io.kubernetes.pod.name:  logtail-ds-44jbg
io.kubernetes.pod.namespace:  kube-system
io.kubernetes.pod.uid:  6ddcf598-1c81-11e9-9ddf-00163e0c7cbe
k8s-app:  logtail-ds
kubernetes.io/cluster-service:  true
name:  k8s_POD_logtail-ds-44jbg_kube-system_6ddcf598-1c81-11e9-9ddf-00163e0c7cbe_0
pod-template-generation:  9
version:  v1.0
```

The following table describes the log fields of Docker events. For more information, see Docker events.

Log field	Description
type	The type of a resource, for example, container or image.
action	The type of an action, for example, destroy or status.
_id_	The unique ID of an event.
_time_nano_	The timestamp of an event.

3.7.9. Collect Windows event logs

Logtail allows you to configure plug-ins to collect Windows event logs. This topic describes how to configure Logtail in the Log Service console to collect Windows event logs.

Prerequisites

Logtail is installed on the server that you use to collect Windows event logs. For more information, see Install Logtail on a Windows server.

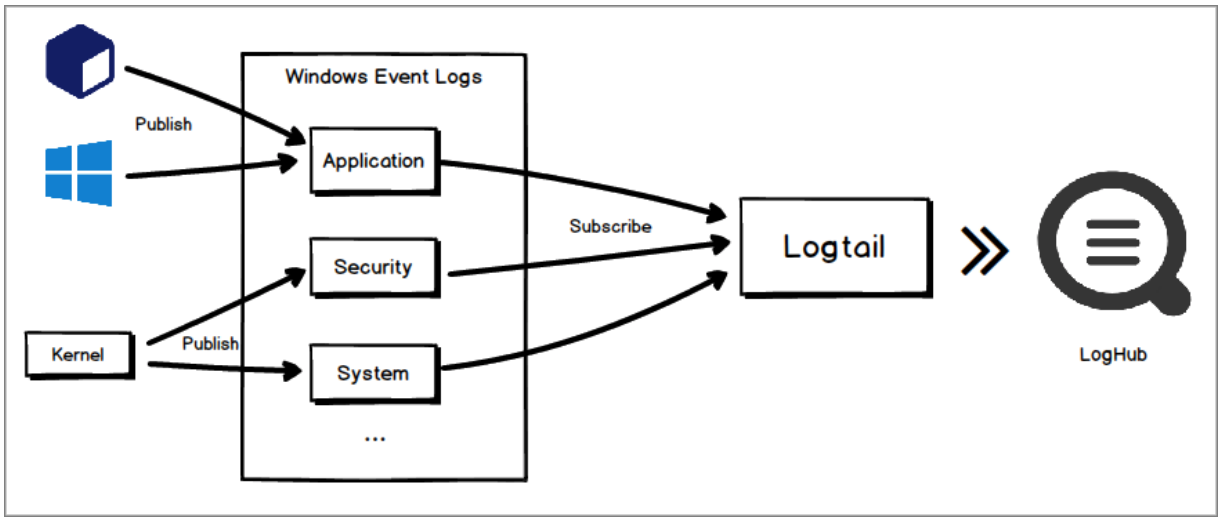
> **Note** Only Windows servers that run Logtail 1.0.0.0 or later are supported.

Implementation

The Windows Event Log API and Event Logging API are provided in Windows operating systems to record event logs. The Event Logging API is an upgrade of the Windows Event Log API, and is provided only in the Windows Vista operating system or later. Logtail plug-ins automatically select an API based on the operating system to obtain Windows event logs. Windows Event Log is preferred.

The publish-subscribe model is used to collect Windows event logs, as shown in the following figure. An application or kernel publishes event logs to a specified channel, such as an application, security, or system channel. Logtail uses the corresponding plug-in to call the Windows Event Log API or Event Logging API to subscribe to these channels. This way, Logtail continuously collects event logs and sends the logs to Log Service.

Logtail allows you to collect logs from multiple channels at the same time. For example, you can collect logs from the application channel and system channel.



View channel information

You can view the information of channels in the **Event Viewer** of a Windows server.

1. Click **Start**.
2. Search for and click **Event Viewer**. The Event Viewer window appears.
3. In the left-side navigation pane, expand **Windows Logs**.
4. View the full names of channels.

   Right-click the destination channel under **Windows Logs** and select **Properties**. In the window that appears, you can view the full name of the channel. The following full names of common channels are displayed under **Windows Logs**.

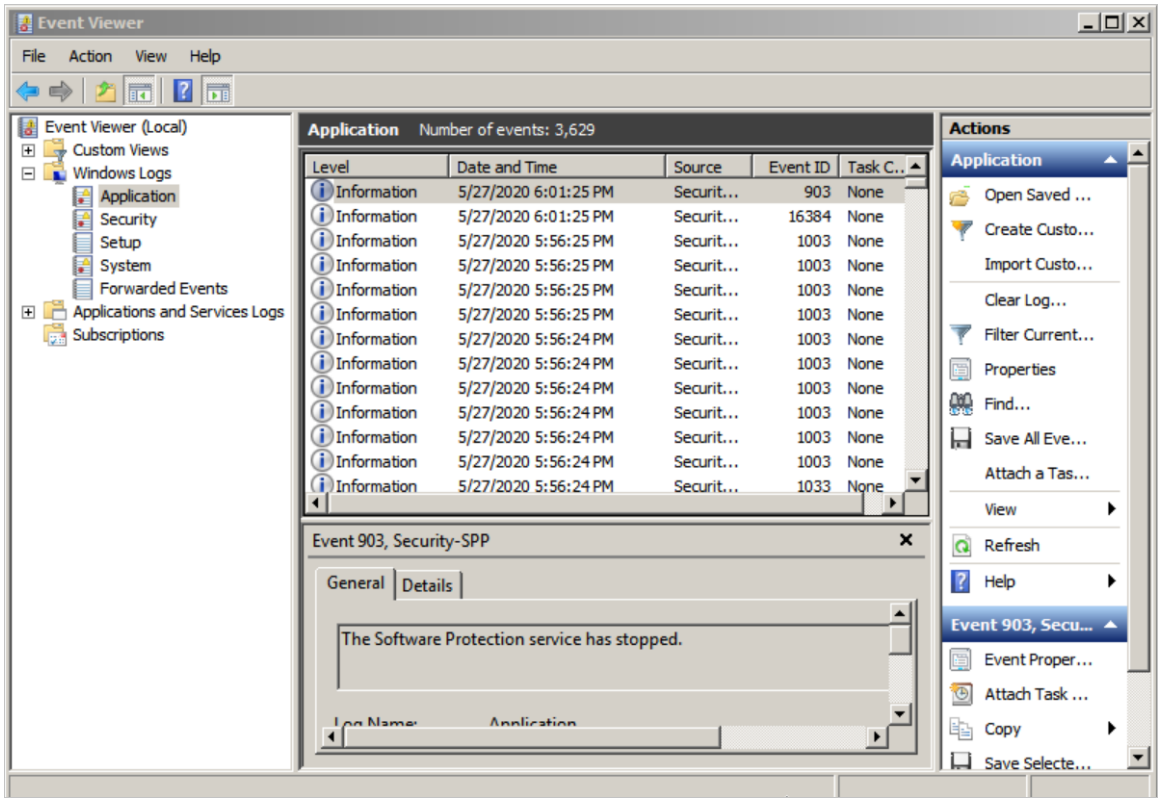
   - Application
   - Security
   - Setup

- System

5. View the information of channels.

   Click the destination channel under **Windows Logs**. In the window that appears, you can view the level, date and time, source, and ID of each event.

   You can set filtering conditions in Logtail configurations based on the preceding fields.



Procedure

1.
2. In the **Import Data** section, select **Custom Data Plug-in**.
3. Select the project and Logstore. Then, click **Next**.
4. In the **Create Machine Group** step, create a machine group.
   - If a machine group is available, click **Using Existing Machine Groups**.
   - If no machine group is available, perform the following steps. These steps take ECS instances as an example to describe how to create a machine group:
     a. Install Logtail on ECS instances. For more information, see Install Logtail on ECS instances.

        If Logtail is installed on the ECS instances, click **Complete Installation**.

        > **Note** If you need to collect logs from user-created clusters or servers of third-party cloud service providers, you must install Logtail on these servers. For more information, see Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

     b. After you install Logtail, click **Complete Installation**.

c. On the page that appears, set relevant parameters for the machine group. For more information, see Create an IP address-based machine group or Create a custom ID-based machine group.

5.

6. In the **Specify Data Source** step, set the **Config Name** and **Plug-in Config** parameters.
   - inputs is required and is used to configure the data collection settings for the Logtail configuration. You must configure inputs based on your data source.

     > **Note** You can specify only one type of data source in inputs.

   - processors is optional and is used to configure the data processing settings for the Logtail configuration. You can specify one or more data processing methods in processors. For more information, see Overview.

   The following example shows how to collect logs from the **Application** and **System** channels. The value of the IgnoreOlder parameter is set to 259200 (three days) to prevent collecting excessive logs.

```
{
    "inputs": [
        {
            "type": "service_wineventlog",
            "detail": {
                "Name": "Application",
                "IgnoreOlder": 259200
            }
        },
        {
            "type": "service_wineventlog",
            "detail": {
                "Name": "System",
                "IgnoreOlder": 259200
            }
        }
    ]
}
```

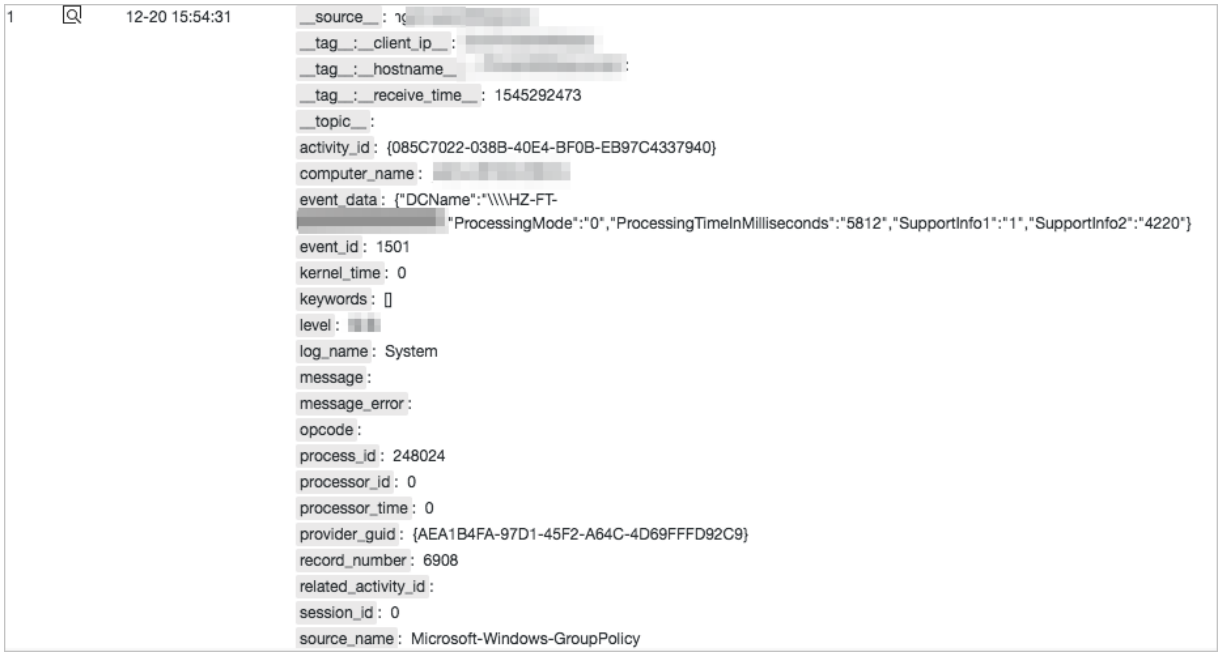
Parameter	Type	Required	Description
type	string	Yes	The type of the data source. Set the value to service_wineventlog.
Name	string	Yes	The name of the channel to which the collected event logs belong. You can specify only one channel. Default value: Application. This value indicates that event logs are collected from the **application** channel. For information about how to obtain the full name of a channel, see Step 4.

Parameter	Type	Required	Description
IgnoreOlder	uint	No	Filters logs by event time. This parameter indicates an offset of the time from which event logs are collected. Unit: seconds. Logs generated earlier than the specified time are not collected. Examples: ◦ 3600: Logs generated 1 hour before the start time of collection are not collected. ◦ 14400: Logs generated 4 hours before the start time of collection are not collected. Default value: null. This value indicates that event logs are not filtered by time during log collection. **Note** This parameter takes effect only when logs are collected for the first time. Logtail records checkpoints during log collection. This mechanism prevents repeated log collection.
Level	string	No	Filters logs by severity. Default value: information, warning, error, critical. This value indicates that logs of all severity levels except verbose are collected. Available severity levels include information, warning, error, critical, and verbose. You can specify multiple severity levels for the parameter and separate these severity levels with commas (,). **Note** This parameter is available only when you call the Windows Event Log API. Therefore, this parameter is used only in the Windows Vista operating system or later.

Parameter	Type	Required	Description
EventID	string	No	Filters logs by event ID. You can specify one or more IDs for positive filtering (specify an event ID or a range) or negative filtering (specify an event ID). Default value: null. This value indicates that all event logs are collected. Examples: ○ 1-200: Only the logs whose event IDs are 1 to 200 are collected. ○ 20: Only the logs whose event IDs are 20 are collected. ○ -100: All logs are collected, except the logs whose event IDs are 100. ○ 1-200,-100: The logs whose event IDs are 1 to 99 or 101 to 200 are collected. You can separate multiple IDs or ID ranges with commas (,). **Note** This parameter is available only when you call the Windows Event Log API. Therefore, this parameter is used only in the Windows Vista operating system or later.
Provider	String array	No	Filters logs by event source. For example, the value ["App1", "App2"] specifies that only event logs from the sources named App1 and App2 are collected. Default value: null. This value indicates that event logs from all sources are collected. **Note** This parameter is available only when you call the Windows Event Log API. Therefore, this parameter is used only in the Windows Vista operating system or later.
IgnoreZeroValue	boolean	No	Filters unavailable fields. Some fields may be unavailable in a log entry. You can set the value of the parameter based on the data type of an unavailable field. For example, set the value to 0 if the data type of an unavailable field is integer. Default value: false. This value indicates that unavailable fields are not filtered.

What's next

After Windows event logs are collected to Log Service, you can view the logs in the Log Service console.



Parameter	Type	Description
activity_id	string	The global transaction ID (GTID) of the event. Events that belong to the same transaction have the same GTID.
computer_name	string	The name of the server where the event occurs.
event_data	JSON object	The data related to the event.
event_id	int	The ID of the event.
kernel_time	int	The kernel time that is consumed by the event. In most cases, the value is 0.
keywords	JSON array	The keyword that is associated with the event. Keywords are used to classify events.
level	string	The severity level of the event.
log_name	string	The name of the channel from which the event is obtained. The value of this parameter is the same as the value of the Name parameter that is specified in the Logtail plug-in configurations.
message	string	The message that is associated with the event.
message_error	string	The error that occurs when the message associated with the event is parsed.
opcode	string	The operation code that is associated with the event.
process_id	int	The process ID of the event.

Parameter	Type	Description
processor_id	int	The ID of the processor that is associated with the event. In most cases, the value is 0.
processor_time	int	The processor time that is consumed by the event. In most cases, the value is 0.
provider_guid	string	The GTID of the event source.
record_number	int	The record number that is associated with the event. The record number increases when an event is written to Log Service. If the number exceeds 2^{32} (Event Logging) or 2^{64} (Windows Event Log), the record number starts from 0 again.
related_activity_id	string	The GTID of another transaction that is associated with the transaction to which the event belongs.
session_id	int	The session ID of the event. In most cases, the value is 0.
source_name	string	The source of the event. The value of this parameter is the same as the value of the Provider parameter that is specified in the Logtail configurations.
task	string	The task that is associated with the event.
thread_id	int	The thread ID of the event.
type	string	The API that is used to obtain the event.
user_data	JSON object	The user data that is associated with the event.
user_domain	string	The user domain that is associated with the event.
user_identifier	string	The Windows Security Identifier (SID) of the user that is associated with the event.
user_name	string	The username that is associated with the event.
user_time	int	The user time that is consumed by the event. In most cases, the value is 0.
user_type	string	The user type that is associated with the event.
version	int	The version of the event.
xml	string	The original information of the event in the XML format.

3.8. Customize Logtail plug-ins to process data

3.8.1. Overview

If you have complex logs that cannot be parsed in basic modes such as regular expression, NGINX, and JSON, you can use Logtail plug-ins to parse logs. You can configure Logtail plug-ins for one or more processing methods. Then, Logtail executes the processing methods in sequence.

Limits

- Performance limits

  When a plug-in is used to process data, Logtail consumes more resources. Most of these resources are CPU resources. You can modify the Logtail parameter settings based on your needs. For more information, see Configure the startup parameters of Logtail. If raw data is generated at a speed higher than 5 MB/s, we recommend that you do not use multiple plug-ins to process the data. You can use a Logtail plug-ins to simplify data processing, and then use the data transformation feature to further process the data.

- Limits on text logs

  Log Service allows you to process text logs in basic modes such as the regular expression, NGINX, or JSON mode. Log Service also allows you to use Logtail plug-ins to process text logs.However, Logtail plug-ins have the following limits on text logs:

  - If you enable the plug-in processing feature, some advanced features of the specified mode become unavailable. For example, you cannot configure the filter, upload raw logs, configure the system time zone, drop logs that fail to be parsed, or upload incomplete log entries (in delimiter mode).
  - Plug-ins use the line mode to process text logs. In this mode, file-level metadata such as __tag__:__path__ and __topic__ is stored in each log entry. If you use Logtail plug-ins to process data, the following limits apply to tag-related features:
    - You cannot use the contextual query and LiveTail features because these features depend on fields such as __tag__:__path__.
    - The name of the __topic__ field is renamed to __log_topic__.
    - Fields such as __tag__:__path__ no longer have original field indexes. You must configure indexes for these fields.

Usage notes

When you configure data processing methods, you must set the key in the configuration file to processors and set the value to an array of JSON objects. Each object of the array contains the details of a processing method.

Each processing method contains the type and detail fields. The type field specifies the type of the processing method and the detail field contains configuration details.

```
"processors" : [
    {
        "type": "processor_split_char",
        "detail": {
            "SourceKey": "content",
            "SplitSep": "|",
            "SplitKeys": [
                "method",
                "type",
                "ip",
                "time",
                "req_id",
                "size",
                "detail"
            ]
        }
    },
    {
        "type": "processor_anchor",
        "detail": {
            "SourceKey": "detail",
            "Anchors": [
                {
                    "Start": "appKey=",
                    "Stop": ",env=",
                    "FieldName": "appKey",
                    "FieldType": "string"
                }
            ]
        }
    }
]
```

The following table describes available Logtail plug-ins and the operations that you can perform by using these plug-ins.

Logtail plug-in	Description
processor_regex	You can extract fields by using the processor_regex plug-in and matching the specified fields based on a regular expression. For more information, see Extract log fields by using a regular expression.
processor_anchor	You can anchor strings and extract fields by using the processor_anchor plug-in, start keyword, and stop keyword. For more information, see Extract log fields by using start and stop keywords.
processor_split_char	You can delimit fields by using the processor_split_char plug-in and a specified single-character delimiter. For more information, see Extract log fields by using a single-character delimiter.
processor_split_string	You can delimit fields by using the processor_split_string plug-in and a specified multi-character delimiter. For more information, see Extract log fields by using a multi-character delimiter.

Logtail plug-in	Description
processor_split_key_value	Use the processor_split_key_value plug-in in key-value pair mode to extract fields. For more information, see Extract log fields by splitting key-value pairs.
processor_add_fields	You can add fields to a log entry by using the processor_add_fields plug-in. For more information, see Add fields.
processor_drop	You can drop specified fields from a log entry by using the processor_drop plug-in. For more information, see Drop fields.
processor_rename	You can rename specified fields by using the processor_rename plug-in. For more information, see Rename columns.
processor_packjson	You can encapsulate one or more fields into a JSON-formatted field by using the processor_packjson plug-in. For more information, see Encapsulate fields.
processor_json	You can expand specified fields from JSON objects by using the processor_json plug-in. For more information, see Expand JSON fields.
processor_filter_regex	You can filter logs by using the processor_filter_regex plug-in. For more information, see Filter logs.
processor_gotime	You can extract time information from a specified field in a time format supported by Golang. Then, you can configure the information as the log time by using the processor_gotime plug-in. For more information, see Time format supported by Go.
processor_strptime	You can extract time information from a specified field in a time format supported by strptime. Then, you can configure the information as the log time by using the processor_strptime plug-in. For more information, see Time format supported by strptime.
processor_geoip	You can transform IP addresses in logs into geo locations by using the processor_geoip plug-in. The geo locations include the country, province, city, longitude, and latitude. For more information, see Transform IP addresses.

3.8.2. Extract fields

You can extract log fields by using regular expressions, start and stop keywords, single-character delimiters, or multi-character delimiters. You can also extract log fields by splitting key-value pairs. This topic describes the parameters of these modes. This topic also provides examples to show how to configure the modes.

Extract log fields by using a regular expression

You can extract log fields by using a regular expression.

- Parameters

  The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_regex.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the source field.
Regex	String	Yes	The regular expression. The fields to be extracted are enclosed in parentheses `()`.
Keys	String array	Yes	The array of fields that are extracted, for example, ["ip", "time", "method"].
NoKeyError	Boolean	No	Specifies whether to report an error if a field is not matched. Default value: false. This value indicates that no error is reported if a field is not matched.
NoMatchError	Boolean	No	Specifies whether to report an error if the regular expression does not match the value of a specified field. Default value: false. This value indicates that no error is reported if a field is not matched.
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: false. This value indicates that the source field is not retained.
FullMatch	Boolean	No	Default value: true. This value indicates that exact match is implemented when the regular expression specified in the Regex parameter attempts to match field values. If you set the value to false, partial match is implemented when the regular expression attempts to match field values.

- Configuration example

  The following example shows how to extract the value of the content field. Then, you can set the names of the destination fields to ip, time, method, url, request_time, request_length, status, length, ref_url, and browser.

  - Raw log entry

    ```
    "content" : "10.200. **. ** - - [10/Aug/2017:14:57:51 +0800] \"POST /PutData?Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature> HTTP/1.1\" 0.024 18204 200 37 \"-\" \"aliyun-sdk-java"
    ```

- Logtail configurations for data processing

```
{
    "type" : "processor_regex",
    "detail" : {"SourceKey" : "content",
         "Regex" : "([\\d\\.] +) \\S+ \\S+ \\[(\\S+) \\S+\\] \"(\\w+) ([^\\\"]*)\" ([\\
d\\.]+) (\\d+) (\\d+) (\\d+|-) \"([^\\\"]*)\" \"([^\\\"]*)\" (\\d+)",
         "Keys"   : ["ip", "time", "method", "url", "request_time", "request_length", "
status", "length", "ref_url", "browser"],
         "NoKeyError" : true,
         "NoMatchError" : true,
         "KeepSource" : false
    }
}
```

- Result

```
"ip" : "10.200. **.**"
"time" : "10/Aug/2017:14:57:51"
"method" : "POST"
"url" : "/PutData? Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C
%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature>"
"request_time" : "0.024"
"request_length" : "18204"
"status" : "200"
"length" : "27"
"ref_url" : "-"
"browser" : "aliyun-sdk-java"
```

Extract log fields by using start and stop keywords

You can use start and stop keywords to anchor strings and extract log fields. If an anchored substring is a JSON string, you can expand the substring and then extract the fields.

- Parameters

  The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_anchor.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the source field.
Anchors	Anchor array	Yes	The list of the parameters that are set to anchor strings.
NoAnchorError	Boolean	No	Specifies whether to report an error if no keyword is found. Default value: false. This value indicates that no error is reported if no keyword is found.
NoKeyError	Boolean	No	Specifies whether to report an error if a field is not matched. Default value: false. This value indicates that no error is reported if a field is not matched.

Parameter	Type	Required	Description
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: false. This value indicates that the source field is not retained.

The following table describes the parameters of the Anchors parameter.

Parameter	Type	Required	Description
Start	String	Yes	The keyword that anchors the start of a substring in a string. If the parameter is not specified, the start of the string is matched.
Stop	String	Yes	The keyword that anchors the end of a substring in a string. If the parameter is not specified, the end of the string is matched.
FieldName	String	Yes	The name of the field to be extracted.
FieldType	String	Yes	The type of the field to be extracted. Valid values: string and json.
ExpondJson	Boolean	No	Specifies whether to expand an anchored JSON substring. Default value: false. This value indicates that an anchored JSON substring is not expanded. This parameter is available only if the value of the FieldType parameter is set to json.
ExpondConnecter	String	No	The character that is used to connect expanded keys. Default value: _.
MaxExpondDepth	Int	No	The maximum depth of JSON expansion. Default value: 0. This value indicates that the depth of JSON expansion is unlimited.

- Configuration example

  The following example shows how to extract the value of the content field. Then, you can set the names of the destination fields to time, val_key1, val_key2, val_key3, value_key4_inner1, and value_key4_inner2.

  - Raw log entry

    ```
    "content" : "time:2017.09.12 20:55:36\tjson:{\"key1\" : \"xx\", \"key2\": false, \"key3\":123.456, \"key4\" : { \"inner1\" : 1, \"inner2\" : false}}"
    ```

- Logtail configurations for data processing

```
{
   "type" : "processor_anchor",
   "detail" : {"SourceKey" : "content",
      "Anchors" : [
          {
              "Start" : "time",
              "Stop" : "\t",
              "FieldName" : "time",
              "FieldType" : "string",
              "ExpondJson" : false
          },
          {
              "Start" : "json:",
              "Stop" : "",
              "FieldName" : "val",
              "FieldType" : "json",
              "ExpondJson" : true
          }
      ]
  }
}
```

- Result

```
"time" : "2017.09.12 20:55:36"
"val_key1" : "xx"
"val_key2" : "false"
"val_key3" : "123.456"
"value_key4_inner1" : "1"
"value_key4_inner2" : "false"
```

Extract log fields by using a single-character delimiter

You can use a specified single-character delimiter to delimit fields. This processing method allows you to specify a quote to enclose the delimiter.

- Parameters

  The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_split_char.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the source field.
SplitSep	String	Yes	The single-character delimiter. The single-character delimiter. You can specify a non-printable character as a single-character delimiter, for example, \u0001.
SplitKeys	String array	Yes	The names of the delimited fields, for example, ["ip", "time", "method"].

Parameter	Type	Required	Description
QuoteFlag	Boolean	No	Specifies whether to use a quote to enclose the specified delimiter. Default value: false. This value indicates that a quote is not used.
Quote	String	No	The quote. The quote must be a single character. You can specify a non-printable character as a quote, for example, \u0001. This parameter is available only if the value of QuoteFlag is set to true.
NoKeyError	Boolean	No	Specifies whether to report an error if a field is not matched. Default value: false. This value indicates that no error is reported if a field is not matched.
NoMatchError	Boolean	No	Specifies whether to report an error if a delimiter is not matched. Default value: false. This value indicates that no error is reported if a delimiter is not matched.
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: false. This value indicates that the source field is not retained.

- Configuration example

  The following example shows how to extract the value of the content field by using a delimiter (|). Then, you can set the names of the destination fields to ip, time, method, url, request_time, request_length, status, length, ref_url, and browser.

  - Raw log entry

    ```
    "content" : "10. **. **. **|10/Aug/2017:14:57:51 +0800|POST|PutData?
    Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C%2028%20Jun%202013%
    2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature>|0.024|18204|200|37|-|
    aliyun-sdk-java"
    ```

  - Logtail configurations for data processing

    ```
    {
       "type" : "processor_split_char",
       "detail" : {"SourceKey" : "content",
          "SplitSep" : "|",
          "SplitKeys" : ["ip", "time", "method", "url", "request_time", "request_length", "
    status", "length", "ref_url", "browser"]
      }
    }
    ```

- Result

```
"ip" : "10. **. **.**"
"time" : "10/Aug/2017:14:57:51 +0800"
"method" : "POST"
"url" : "/PutData? Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C
%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature>"
"request_time" : "0.024"
"request_length" : "18204"
"status" : "200"
"length" : "27"
"ref_url" : "-"
"browser" : "aliyun-sdk-java"
```

Extract log fields by using a multi-character delimiter

You can use a specified multi-character delimiter to delimit fields. You cannot specify a quote to enclose the delimiter.

- Parameters

  The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_split_string.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the source field.
SplitSep	String	Yes	The multi-character delimiter. The multi-character delimiter. You can specify multiple non-printable characters as a multi-character delimiter, for example, \u0001\u0002.
SplitKeys	String array	Yes	The names of the delimited fields, for example, ["key1", "key2"...].
PreserveOthers	Boolean	No	Specifies whether to retain excess fields if the number of fields is greater than the number of fields specified by the SplitKeys parameter. Default value: false. This value indicates that excess fields are not retained.
ExpandOthers	Boolean	No	Specifies whether to parse excess fields. Default value: false. This value indicates that excess fields are not parsed.
ExpandKeyPrefix	String	No	The name prefix of excess fields. For example, if you specify expand_ for the parameter, the first two excess fields are named expand_1 and expand_2.

Parameter	Type	Required	Description
NoKeyError	Boolean	No	Specifies whether to report an error if a field is not matched. Default value: false. This value indicates that no error is reported if a field is not matched.
NoMatchError	Boolean	No	Specifies whether to report an error if a delimiter is not matched. Default value: false. This value indicates that no error is reported if a delimiter is not matched.
KeepSource	Boolean	No	Specifies whether to retain the source field. This value indicates that the source field is not retained.

- Configuration example

  The following example shows how to extract the value of the content field by using a delimiter (|#|). Then, you can set the names of the destination fields to ip, time, method, url, request_time, request_length, status, expand_1, expand_2, and expand_3.

  - Raw log entry

    ```
    "content" : "10. **. **. **|#|10/Aug/2017:14:57:51 +0800|#|POST|#|PutData?
    Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C%2028%20Jun%202013%
    2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature>|#|0.024|#|18204|#|200|#|27|#|-
    |#|
    aliyun-sdk-java"
    ```

  - Logtail configurations for data processing

    ```
    {
       "type" : "processor_split_string",
       "detail" : {"SourceKey" : "content",
          "SplitSep" : "|#|",
          "SplitKeys" : ["ip", "time", "method", "url", "request_time", "request_length", "
    status"],
          "PreserveOthers" : true,
          "ExpandOthers" : true,
          "ExpandKeyPrefix" : "expand_"
      }
    }
    ```

- Result

```
"ip" : "10. **. **.**"
"time" : "10/Aug/2017:14:57:51 +0800"
"method" : "POST"
"url" : "/PutData? Category=YunOsAccountOpLog&AccessKeyId=<yourAccessKeyId>&Date=Fri%2C
%2028%20Jun%202013%2006%3A53%3A30%20GMT&Topic=raw&Signature=<yourSignature>"
"request_time" : "0.024"
"request_length" : "18204"
"status" : "200"
"expand_1" : "27"
"expand_2" : "-"
"expand_3" : "aliyun-sdk-java"
```

Extract log fields by splitting key-value pairs

You can split key-value pairs to extract log fields.

- Parameters

  The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_split_key_value.

  > **Note** Only Logtail V0.16.26 or later supports the plug-in.

Parameter	Type	Required	Description
SourceKey	string	Yes	The name of the source field.
Delimiter	string	No	The delimiter between key-value pairs. Default value: `\t`.
Separator	string	No	The delimiter used to separate the key and the value in a single key-value pair. Default value: :.
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: true. This value indicates that the source field is retained.
ErrIfSourceKeyNotFound	Boolean	No	Specifies whether to trigger an alert if a field is not matched. Default value: true. This value indicates that an alert is triggered if a field is not matched.
DiscardWhenSeparatorNotFound	Boolean	No	Specifies whether to drop the key-value pair if a field is not matched. Default value: false. This value indicates that the key-value pair is not dropped if a field is not matched.

Parameter	Type	Required	Description
ErrIfSeparatorNotFound	Boolean	No	Specifies whether to trigger an alert if the specified delimiter (Separator) does not exist. Default value: true. This value indicates that an alert is triggered if the specified delimiter does not exist.

- Configuration example

  The following example shows how to split the key-value pair of the content field. The delimiter used to separate key-value pairs is a tab character (`/t`). The delimiter used to separate the key and the value in a single key-value pair is a colon (:).

  - Raw log entry

    ```
    "content": "class:main\tuserid:123456\tmethod:get\tmessage:\"wrong user\""
    ```

  - Logtail configurations for data processing

    ```
    {
      "processors":[
        {
          "type":"processor_split_key_value",
          "detail": {
            "SourceKey": "content",
            "Delimiter": "\t",
            "Separator": ":",
            "KeepSource": true
          }
        }
      ]
    }
    ```

  - Result

    ```
    "content": "class:main\tuserid:123456\tmethod:get\tmessage:\"wrong user\""
    "class": "main"
    "userid": "123456"
    "method": "get"
    "message": "\"wrong user\""
    ```

3.8.3. Add fields

You can add fields to a log entry by using the processor_add_fields plug-in. This topic describes the parameters of the processor_add_fields plug-in. This topic also provides examples to show how to configure the plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_add_fields.

> **Note** Only Logtail V0.16.28 or later supports the plug-in.

Parameter	Type	Required	Description
Fields	Map	No	The key-value pairs to be added. You can specify multiple key-value pairs in the parameter.
IgnoreIfExist	Boolean	No	Specifies whether to retain key-value pairs that have the same key. Default value: false. This value indicates that a key-value pair is not retained if the key is the same as another specified key.

Configuration example

The following example shows how to add the aaa2 and aaa3 fields to a log entry.

- Raw log entry

```
"aaa1":"value1"
```

- Logtail configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_add_fields",
      "detail": {
        "Fields": {
          "aaa2": "value2",
          "aaa3": "value3"
        }
      }
    }
  ]
}
```

- Result

```
"aaa1":"value1"
"aaa2":"value2"
"aaa3":"value3"
```

3.8.4. Drop fields

You can drop specified fields from a log entry by using the processor_drop plug-in This topic describes the parameters of the processor_drop plug-in. This topic also provides examples to show how to configure the plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_drop.

> **Note** Only Logtail V0.16.28 or later supports the plug-in.

Parameter	Type	Required	Description
DropKeys	String array	No	The fields to be dropped. You can drop one or more fields from a log entry.

Configuration example

The following example shows how to drop the aaa1 and aaa2 fields from a log entry.

- Raw log entry

```
"aaa1":"value1"
"aaa2":"value2"
"aaa3":"value3"
```

- Logtail configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_drop",
      "detail": {
        "DropKeys": ["aaa1","aaa2"]
      }
    }
  ]
}
```

- Result

```
"aaa3":"value3"
```

3.8.5. Rename columns

You can rename specified fields by using the processor_rename plug-in. This topic describes the parameters of the processor_rename plug-in. This topic also provides examples to show how to configure the plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_rename.

> **Note** Only Logtail V0.16.28 or later supports the plug-in.

Parameter	Type	Required	Description
NoKeyError	Boolean	Yes	Specifies whether to report an error if a field to be renamed is not matched. Default value: false. This value indicates that no error is reported if a field to be renamed is not matched.
SourceKeys	String array	Yes	The source fields to be renamed.
DestKeys	String array	Yes	The renamed fields.

Configuration example

The following example shows how to rename the aaa1 field to bbb1 and the aaa2 field to bbb2.

- Raw log entry

```
"aaa1":"value1"
"aaa2":"value2"
"aaa3":"value3"
```

- Logtail configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_rename",
      "detail": {
        "SourceKeys": ["aaa1","aaa2"],
        "DestKeys": ["bbb1","bbb2"],
        "NoKeyError": true
      }
    }
  ]
}
```

- Result

```
"bbb1":"value1"
"bbb2":"value2"
"aaa3":"value3"
```

3.8.6. Encapsulate fields

You can encapsulate one or more fields into a JSON-formatted field by using the processor_packjson plug-in. This topic describes the parameters of the processor_packjson plug-in. This topic also provides examples to show how to configure the plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_packjson.

> **Note** Only Logtail V0.16.28 or later supports the plug-in.

Parameter	Type	Required	Description
SourceKeys	String array	Yes	The string array-formatted field to be encapsulated.
DestKey	String	No	The destination JSON-formatted field.
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: true. This value indicates that the source field is retained.
AlarmIfIncomplete	Boolean	No	Specifies whether to trigger an alert if the source field is not found. Default value: true. This value indicates that an alert is triggered if the source field is not found.

Configuration example

The following example shows how to encapsulate the a and b fields into the d_key field.

- Raw log entry

```
"a":"1"
"b":"2"
```

- Logtail configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_packjson",
      "detail": {
        "SourceKeys": ["a","b"],
        "DestKey":"d_key",
        "KeepSource":true,
        "AlarmIfEmpty":true
      }
    }
  ]
}
```

- Result

```
"a":"1"
"b":"2"
"d_key":"{\"a\":\"1\",\"b\":\"2\"}"
```

3.8.7. Expand JSON fields

You can use the processor_json plug-in to expand JSON fields. This topic describes the parameters of the processor_json plug-in. This topic also provides an example on how to configure the plug-in.

Parameters

If you set type to processor_json, you can configure the parameters in detail based on the following table.

> **Note** Only Logtail V0.16.28 and later support the processor_json plug-in.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the raw field.
NoKeyError	Boolean	No	Specifies whether to report an error if the raw field is not matched. If you do not include this parameter in the configuration, the value true is used by default. This value indicates that an error is reported if the raw field is not matched.
ExpandDepth	Int	No	The depth of JSON expansion. If you do not include this parameter in the configuration, the value 0 is used by default. This value indicates the depth of JSON expansion is unlimited. If the value is n, the depth of JSON expansion is n.
ExpandConnector	String	No	The character that is used to connect expanded keys. You can leave this parameter empty. If you do not include this parameter in the configuration, an underscore (_) is used by default.
Prefix	String	No	The prefix that is added to expanded keys. If you do not include this parameter in the configuration, this parameter is considered empty.
KeepSource	Boolean	No	Specifies whether to retain the raw field. If you do not include this parameter in the configuration, the value true is used by default. This value indicates that the raw field is retained.
UseSourceKeyAsPrefix	Boolean	No	Specifies whether to add the name of the raw field as a prefix to all expanded keys. If you do not include this parameter in the configuration, the value false is used by default. This value indicates that the name of the raw field is not added.

Parameter	Type	Required	Description
KeepSourceIfParseError	Boolean	No	Specifies whether to retain the raw log if the log fails to be parsed. If you do not include this parameter in the configuration, the value true is used by default. This value indicates that the raw log is retained.

Configuration example

The following example shows how to expand the JSON field s_key and add j and the name of the raw field s_key as a prefix to the expanded keys.

- Raw log

```
"s_key":"{\"k1\":{\"k2\":{\"k3\":{\"k4\":{\"k51\":\"51\",\"k52\":\"52\"},\"k41\":\"41\"}}
})"
```

- Logtail plug-in configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_json",
      "detail": {
        "SourceKey": "s_key",
        "NoKeyError":true,
        "ExpandDepth":0,
        "ExpandConnector":"-",
        "Prefix":"j",
        "KeepSource": false,
        "UseSourceKeyAsPrefix": true
      }
    }
  ]
}
```

- Result

```
"js_key-k1-k2-k3-k4-k51":"51"
"js_key-k1-k2-k3-k4-k52":"52"
"js_key-k1-k2-k3-k41":"41"
```

3.8.8. Filter logs

You can filter logs by using the processor_filter_regex plug-in. This topic describes the parameters that you must configure for the processor_filter_regex plug-in. This topic also provides examples on how to configure the parameters.

Parameters

The following table describes the parameters that you can configure in the detail parameter when you set the type parameter to processor_filter_regex.

> **Note** A log entry is collected only when it exactly matches the regular expressions that are specified in the Include parameter and does not match the regular expressions that are specified in the Exclude parameter.

Parameter	Type	Required	Description
Include	JSON Object	No	An array of key-value pairs. In each key-value pair, the key specifies a field, and the value specifies a regular expression that the value of the same field in each log entry must match. The specified keys are in AND relations. If the values of all fields in a log entry match the regular expressions that are specified in the Include parameter, the log entry is collected.
Exclude	JSON Object	No	An array of key-value pairs. In each key-value pair, the key specifies a field, and the value specifies a regular expression that the value of the same field in each log entry must match. The specified keys are in OR relations. If the value of any field in a log entry matches a regular expression that are specified in the Include parameter, the log entry is not collected.

Configuration example

In this example, only log entries in which the value of the ip field is prefixed by 10, the value of the method field is POST, and the value of the browser field is not aliyun.* are collected.

- Raw logs
  - Log entry 1

    ```
    "ip" : "10.**.**.**"
    "method" : "POST"
    "browser" : "aliyun-sdk-java"
    ```

  - Log entry 2

    ```
    "ip" : "10.**.**.**"
    "method" : "POST"
    "browser" : "chrome"
    ```

  - Log entry 3

    ```
    "ip" : "192.168.*.*"
    "method" : "POST"
    "browser" : "ali-sls-ilogtail"
    ```

- Logtail plug-in configuration for processing

```
{
   "type" : "processor_filter_regex",
    "detail" : {
         "Include" : {
            "ip" : "10\\..*",
            "method" : "POST"
         },
         "Exclude" : {
            "browser" : "aliyun.*"
         }
    }
}
```

- Output data

Log entry	Collected	Reason
Log entry 1	No	The value of the browser parameter matches a regular expression that is specified in the Exclude parameter.
Log entry 2	Yes	The values of all fields match the regular expressions that are specified in the Include parameter.
Log entry 3	No	The value of the ip parameter does not match the regular expression that is specified in the Include parameter.

3.8.9. Extract log time

You can parse the time field in raw logs by using the processor_gotime plug-in or the processor_strptime plug-in. This topic describes the parameters of the processor_gotime plug-in and the processor_strptime plug-in. This topic also provides examples on how to configure the plug-ins.

Time format supported by Go

The processor_gotime plug-in parses the time field in raw logs into the time format supported by Go. You can configure the parsed time value as log time in Log Service. For more information, see Go.

- Parameters

  The following table describes the parameters that you can configure in detail if you set the type parameter to processor_gotime.

  > **Note** Only Logtail V0.16.28 and later support the processor_gotime plug-in.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the raw field.
SourceFormat	String	Yes	The raw time format.

Parameter	Type	Required	Description
SourceLocation	Int	Yes	The raw time zone. If the parameter is empty, the time zone of the server or container on which Logtail runs is used.
DestKey	String	Yes	The name of the parsed field.
DestFormat	String	Yes	The parsed time format.
DestLocation	Int	No	The parsed time zone. If the parameter is empty, the time zone of the server on which Logtail runs is used.
SetTime	Boolean	No	Specifies whether to configure the parsed time value as log time. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that the parsed time value is configured as log time.
KeepSource	Boolean	No	Specifies whether to retain the raw field. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that the raw field is retained.
NoKeyError	Boolean	No	Specifies whether to report an error if the raw field is not found. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that an error is reported if the raw field is not found.
AlarmIfFail	Boolean	No	Specifies whether to trigger an alert if log time fails to be extracted. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that an alert is triggered if log time fails to be extracted.

- Example

  In this example, the raw time value in the `2006-01-02 15:04:05 (UTC+8)` format is extracted from the s_key field and parsed into the format `2006/01/02 15:04:05 (UTC+9)`. The parsed time value is configured as log time in Log Service and stored in the d_key field.

  - Raw log

    ```
    "s_key":"2019-07-05 19:28:01"
    ```

- Logtail plug-in configuration for data processing

```
{
  "processors":[
    {
      "type":"processor_gotime",
      "detail": {
        "SourceKey": "s_key",
        "SourceFormat":"2006-01-02 15:04:05",
        "SourceLocation":8,
        "DestKey":"d_key",
        "DestFormat":"2006/01/02 15:04:05",
        "DestLocation":9,
        "SetTime": true,
        "KeepSource": true,
        "NoKeyError": true,
        "AlarmIfFail": true
      }
    }
  ]
}
```

- Results

```
"s_key":"2019-07-05 19:28:01"
"d_key":"2019/07/05 20:28:01"
```

Time format supported by strptime

The processor_strptime plug-in parses the time field in raw logs into the time format supported by strptime. You can configure the parsed time value as log time in Log Service. For more information, see strptime.

- Parameters

  The following table describes the parameters that you can configure in detail if you set the type parameter to processor_strptime.

  > **Note** Only Logtail V0.16.28 and later support the processor_strptime plug-in.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the raw field.
Format	String	Yes	The raw time format.
AdjustUTCOffset	Boolean	No	Specifies whether to change the time zone. If you do not include this parameter in the configuration, the system uses the value false by default. The value false indicates that the time zone is not changed.

Parameter	Type	Required	Description
UTCOffset	Int	No	The offset that you want to use to change the time zone. Unit: seconds. For example, the value 28800 indicates that the time zone is changed to UTC+8.
AlarmIfFail	Boolean	No	Specifies whether to trigger an alert if log time fails to be extracted. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that an alert is triggered if log time fails to be extracted.
KeepSource	Boolean	No	Specifies whether to retain the raw field. If you do not include this parameter in the configuration, the system uses the value true by default. The value true indicates that the raw field is retained.

Parameter	Type	Required	Description
EnablePreciseTimestamp	Boolean	No	Specifies whether to extract time values with high precision. If you do not include this parameter in the configuration, the system uses the value false by default. The value false indicates that time values with high precision are not extracted. If you set this parameter to true, the processor_strptime plug-in parses the value of the SourceKey field into a timestamp with millisecond precision and stores the timestamp in the field specified by the PreciseTimestampKey parameter. **Note** ◦ Before you set this parameter to true, make sure that the value of the SourceKey field uses one of the following time precisions: ms, us, and ns. ◦ Only Logtail V1.0.32 and later support this parameter.
PreciseTimestampKey	String	No	The field that stores timestamps with high precision. If you do not include this parameter in the configuration, the system uses the precise_timestamp field by default.
PreciseTimestampUnit	String	No	The unit of the timestamp with high precision. If you do not include this parameter in the configuration, the system uses ms by default. Valid values: ms, us, and ns.

- Examples

  In the following examples, the time value in the `%Y/%m/%d %H:%M:%S` format is extracted from the log_time field and parsed into the log time in the format that you specify. The time zone of your server is used.

- Example 1: The time zone is UTC+8.
  - Raw log

    ```
    "log_time":"2016/01/02 12:59:59"
    ```

  - Logtail plug-in configuration for data processing

    ```
    {
      "processors":[
        {
          "type":"processor_strptime",
          "detail": {
            "SourceKey": "log_time",
            "Format": "%Y/%m/%d %H:%M:%S"
          }
        }
      ]
    }
    ```

  - Results

    ```
    "log_time":"2016/01/02 12:59:59"
    Log.Time = 1451710799
    ```

- Example 2: The time zone is UTC+7.
  - Raw log

    ```
    "log_time":"2016/01/02 12:59:59"
    ```

  - Logtail plug-in configuration for data processing

    ```
    {
      "processors":[
        {
          "type":"processor_strptime",
          "detail": {
            "SourceKey": "log_time",
            "Format": "%Y/%m/%d %H:%M:%S",
            "AdjustUTCOffset": true,
            "UTCOffset": 25200
          }
        }
      ]
    }
    ```

  - Results

    ```
    "log_time":"2016/01/02 12:59:59"
    Log.Time = 1451714399
    ```

- Example 3: The time zone is UTC +7.
  - Raw log

```
"log_time":"2016/01/02 12:59:59.123"
```

  - Logtail plug-in configuration for data processing

```
{
  "processors":[
    {
      "type":"processor_strptime",
      "detail": {
        "SourceKey": "log_time",
        "Format": "%Y/%m/%d %H:%M:%S.%f",
        "EnablePreciseTimestamp": true
      }
    }
  ]
}
```

  - Results

```
"log_time":"2016/01/02 12:59:59.123"
"precise_timestamp": 1451714399123
Log.Time = 1451714399
```

- Common time expressions

> **Note** The processor_strptime plug-in can parse time values in the %f format. %f indicates the fractional part of the second. The highest precision that is supported by the processor_strptime plug-in is the nanosecond.

Example	Time expression
2016/01/02 12:59:59	%Y/%m/%d %H:%M:%S
2016/01/02 12:59:59.1	%Y/%m/%d %H:%M:%S.%f
2016/01/02 12:59:59.987654321 +0700 (UTC)	%Y/%m/%d %H:%M:%S.%f %z (%Z)
2016/Jan/02 12:59:59,123456	%Y/%b/%d %H:%M:%S,%f
2019-07-15T04:16:47:123Z	%Y-%m-%dT%H:%M:%S:%f

3.8.10. Transform IP addresses

You can transform IP addresses in logs into geo locations by using the processor_geoip plug-in. The geo locations include the country, province, city, longitude, and latitude. This topic describes the parameters of the processor_geoip plug-in. This topic also provides examples to show how to configure the plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter when you set the type parameter to processor_geoip.

> **Note**
>
> - GeoIP databases are not included in the Logtail installation package. You must download and configure a GeoIP database on a local server. We recommend that you download a database that provides the city information of an IP address. For more information, see GeoLite2 Free Geolocation Data.
> - Make sure that the database format is MMDB.

Parameter	Type	Required	Description
SourceKey	String	Yes	The name of the source field to be converted.
DBPath	String	Yes	The absolute path of the GeoIP database, for example, */user/data/GeoLite2-City_20180102/GeoLite2-City.mmdb*.
NoKeyError	Boolean	No	Specifies whether to report an error if a field is not matched. Default value: false. This value indicates that no error is reported if a field is not matched.
NoMatchError	Boolean	No	Specifies whether to report an error if an IP address is invalid or is not matched in the database. Default value: false. This value indicates that no error is reported if an IP address is invalid or is not matched in the database.
KeepSource	Boolean	No	Specifies whether to retain the source field. Default value: true. This value indicates that the source field is retained.
Language	String	No	The language of the GeoIP database. Default value: zh-CN. Make sure that your GeoIP database can be displayed in a language that is suitable for your business.

Configuration example

The following example shows how to configure the processing method to process IP addresses in logs.

- Raw log entry

```
"source_ip" : "**. **. **. **"
```

- Logtail configurations for data processing

```
{
   "type": "processor_geoip",
    "detail": {
         "SourceKey": "ip",
         "NoKeyError": true,
         "NoMatchError": true,
         "KeepSource": true,
         "DBPath" : "/user/local/data/GeoLite2-City_20180102/GeoLite2-City.mmdb"
    }
}
```

- Result

```
"source_ip_city_" : "**. **. **.**"
"source_ip_province_": "Zhejiang"
"source_ip_city_": "Hangzhou"
"source_ip_province_code_" : "ZJ"
"source_ip_country_code_" : "CN"
"source_ip_longitude_" : "120.********"
"source_ip_latitude_" : "30. ********"
```

3.8.11. Append data to a field

You can append specific data to an existing field or a field that does not exist by using the processor_appender plug-in. You can also add template variables to the value of a field. In most cases, this plug-in is used together with plug-ins that monitor time series data, such as input_prometheus and input_system_v2. This way, you can append specific data to Prometheus data that is pulled.

Notice Only Logtail V0.16.66 or later supports the processor_appender plug-in.

Parameters

The following table describes the parameters that you can specify in the detail parameter if you set the type parameter to *processor_appender*.

Plug-in parameters

Parameter	Type	Required	Description
Key	string	Yes	The name of the field.
Value	string	Yes	The data that you want to append to the field. Log Service allows you to add template variables to the value of a field. For more information, see Template variables.

Parameter	Type	Required	Description
SortLabels	boolean	No	If you want to add the __labels__ field, you must set the Key parameter to *__labels__*. Then, you must set the SortLabels parameter to *true* to sort the appended labels in alphabetical order. Otherwise, disordered labels cause errors when queries are performed. Default value: false.

Template variables

Template variable	Description	Configuration example	Sample result
{{__ip__}}	This variable is replaced by the IP address of the server where Logtail resides.	"Value": "{{__ip__}}"	"Value": "192.0.2.1"
{{__host__}}	This variable is replaced by the hostname of the server where Logtail resides.	"Value": "{{__host__}}"	"Value": "logtail-ds-xdfaf"
{{$xxxx}}	This variable is used to reference an environment variable and must start with a dollar sign ($). This variable is replaced by the value of the environment variable.	"Value": "{{$WORKING_GROUP}}"	"Value": "prod"

Example

The IP address of the server where Logtail resides is 192.0.2.1, the hostname is david, and the value of the environment variable WORKING_GROUP is prod. The following example shows how to append the preceding data to the __labels__ field:

- Raw log entry

```
"__labels__":"a#$#b"
```

- Logtail plug-in configurations for data processing

```
{
  "processors":[
    {
      "type":"processor_appender",
      "detail": {
        "Key": "__labels__",
        "Value": "|host#$#{{__host__}}|ip#$#{{__ip__}}|group#$#{{$WORKING_GROUP}}",
        "SortLabels": true
      }
    }
  ]
}
```

- Result

```
"__labels__":"a#$#b|group#$#prod|host#$#david|ip#$#192.0.2.1"
```

3.9. Use built-in alert monitoring rules for Logtail

Log Service provides built-in alert monitoring rules. If you want to monitor Logtail in real time, you must enable the alert instances of the related alert monitoring rules. You can receive alert notifications based on the notification method that you specify, for example, DingTalk. This topic describes how to use built-in alert monitoring rules for Logtail.

Prerequisites

The important log feature is enabled for the project that you want to manage. For more information, see Enable the service log feature.

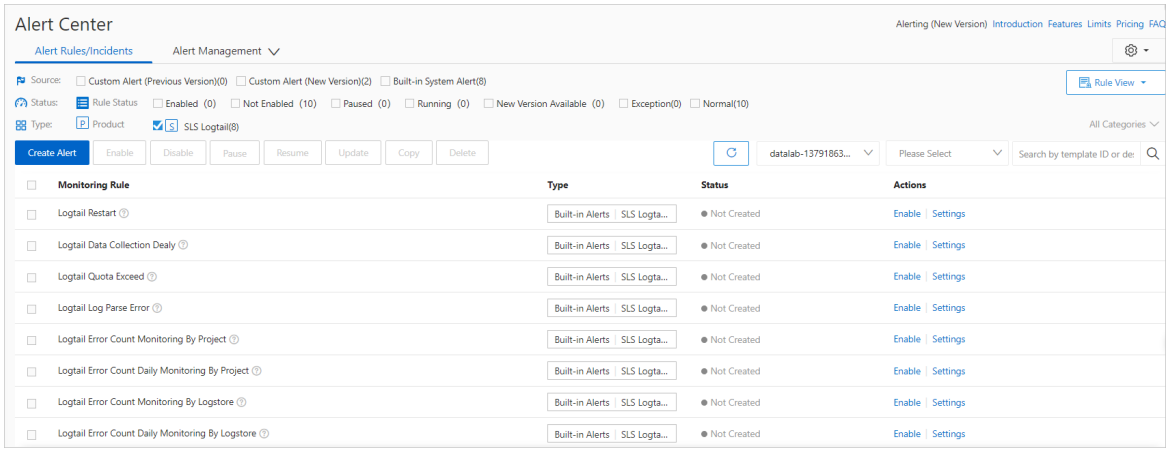
Context

After you enable the important log feature for the project, Log Service automatically creates a Logstore named **internal-diagnostic_log** in the project to store Logtail heartbeat logs. Log Service presets alert monitoring rules based on Logtail heartbeat logs. You can use the built-in alert monitoring rules to monitor Logtail in real time.

Procedure

1.
2. In the Projects section, find the project that you want to manage and click the name of the project.

   You must select the project for which you enable the important log feature.
3. In the left-side navigation pane, click **Alerts**.
4. On the **Alert Rules/Incidents** tab, click **SLS Logtail**.



5. In the alert monitoring rule list, find the alert monitoring rule that you want to use and click **Enable** in the Actions column.

   The default values of the parameters in each alert monitoring rule are specified. You can click **Enable** without the need to configure an alert monitoring rule. If you want to modify parameters, click **Settings**. For more information, see Alert monitoring rules for Logtail.

Alert monitoring rules for Logtail

Log Service provides the following built-in alert monitoring rules to monitor Logtail:

- Logtail Restart
- Logtail Data Collection Delay
- Logtail Quota Exceed
- Logtail Log Parse Error
- Logtail Error Count Monitoring By Project
- Logtail Error Count Daily Monitoring By Project
- Logtail Error Count Monitoring By Logstore
- Logtail Error Count Daily Monitoring By Project

- Logtail Restart

Item	Description
Functionality	This rule is used to monitor the restart behavior of Logtail.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If the number of times that a Logtail client restarts exceeds the specified threshold within the previous 5 minutes, an alert is triggered.

Item	Description
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Threshold(Critical)**: If the number of times that a Logtail client restarts is greater than this threshold within the previous 5 minutes, an alert whose severity level is Critical is triggered. Default value: 3. ○ **Threshold(High)**: If the number of times that a Logtail client restarts is greater than this threshold within the previous 5 minutes, an alert whose severity level is High is triggered. Default value: 1.

- Logtail Data Collection Delay

Item	Description
Functionality	This rule is used to check whether latency occurs when Logtail collects data.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If latency occurs when Logtail collects data for a Logstore within the previous 5 minutes, an alert is triggered.
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium.

- Logtail Quota Exceed

Item	Description
Functionality	This rule is used to check whether the Logtail quota is exceeded.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If Logtail fails to send data to a Logstore within the previous 5 minutes because the quota is exhausted, an alert is triggered.

Item	Description
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium.

- Logtail Log Parse Error

Item	Description
Functionality	This rule is used to monitor the exceptions that occur when Logtail parses logs.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If an error occurs when Logtail parses logs for a Logstore within the previous 5 minutes, an alert is triggered.
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium.

- Logtail Error Count Monitoring By Project

Item	Description
Functionality	This rule is used to monitor the number of Logtail collection errors.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If the number of Logtail collection errors that occur in a project exceeds the specified threshold within the previous 5 minutes, an alert is triggered.

Item	Description
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium. ○ **Threshold**: If the number of Logtail collection errors that occur in a project is greater than this threshold within the previous 5 minutes, an alert is triggered.

- Logtail Error Count Daily Monitoring By Project

Item	Description
Functionality	This rule is used to monitor the daily changes in the number of Logtail collection errors within a specific period of time.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If the daily growth rate of the number of Logtail collection errors that occur in a project exceeds the specified threshold within the previous 5 minutes, an alert is triggered.
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium. ○ **Threshold**: If the daily growth rate of Logtail collection errors that occur in a project is greater than this threshold within the previous 5 minutes, an alert is triggered.

- Logtail Error Count Monitoring By Logstore

Item	Description
Functionality	This rule is used to monitor the number of Logtail collection errors.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If the number of Logtail collection errors that occur in a Logstore exceeds the specified threshold within the previous 5 minutes, an alert is triggered.

Item	Description
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium. ○ **Threshold**: If the number of Logtail collection errors that occur in a Logstore is greater than this threshold within the previous 5 minutes, an alert is triggered.

- Logtail Error Count Daily Monitoring By Logstore

Item	Description
Functionality	This rule is used to monitor the daily changes in the number of Logtail collection errors within a specific period of time.
Detection frequency and detection time range	The data that is generated within the previous 5 minutes is checked every 5 minutes.
Trigger condition	If the daily growth rate of the number of Logtail collection errors that occur in a Logstore exceeds the specified threshold within the previous 5 minutes, an alert is triggered.
Parameter settings	○ **Action Policy**: The action policy that is associated with the current alert monitoring rule. Log Service sends alert notifications to the specified users based on the action policy. The default action policy is sls.app.logtail.builtin. You can modify the built-in action policy or create an action policy based on your business requirements. For more information, see Create an action policy. ○ **Severity**: The severity level of the alert. Valid values: Critical, High, Medium, Low, and Report. Default value: Medium. ○ **Threshold**: If the daily growth rate of Logtail collection errors that occur in a Logstore is greater than this threshold within the previous 5 minutes, an alert is triggered.

3.10. Logtail limits

This topic describes the limits of Logtail, including the limits on log files, checkpoints, Logtail configurations, resources, performance metrics, and troubleshooting.

Limits on log files

Item	Description

Item	Description
Log file encoding	Logtail supports log files that are encoded in UTF-8 and GBK. We recommend that you use UTF-8-encoded log files to improve processing performance. If log files are encoded in other formats, errors such as garbled characters and data loss may occur.
Log file size	The log file size is unlimited.
Log file rotation	Logtail supports log file rotation. Both `.log*` and `.log` files are supported in rotation.
Log collection behavior performed when log parsing is blocked	When log parsing is blocked, Logtail keeps the log file descriptor (FD) open. If log file rotation occurs multiple times during the blocking period, Logtail attempts to ensure that new log files are parsed in sequence. If the number of new files that are not parsed exceeds 20, Logtail does not process the excess log files.
Symbolic link	Monitored directories can be symbolic links.
Log size	The maximum size of a log is 512 KB. If a multi-line log is split by using a regular expression to match the first line, the maximum size of each log after splitting is still 512 KB. If the size of a log exceeds 512 KB, the log is forcibly split into multiple parts for collection. For example, if the size of a log is 1,025 KB, the log is split into three parts: 512 KB, 512 KB, and 1 KB. Then, the log parts are collected in sequence.
Regular expression	Logtail uses regular expressions that are compatible with Perl.
Multiple Logtail configurations for the same log file	You cannot use multiple Logtail configurations for the same log file. We recommend that you store data collected from a log file to the same Logstore. You can configure multiple tasks to subscribe to logs collected from different files. If you want to use multiple Logtail configurations for the same log file, configure symbolic links for log files to bypass this limit.
File opening behavior	When Logtail collects data from a log file, Logtail keeps the log file open. If the log file is not modified for more than 5 minutes and log rotation does not occur, Logtail closes the log file.
First log collection behavior	Logtail collects data only from incremental log files. If the size of a log file exceeds 1 MB the first time the modification to the log file is detected, Logtail collects data from the last 1 MB. If the log file size does not exceed 1 MB, Logtail collects data from the beginning of the log file. If the log file is not modified after the Logtail configuration is delivered, Logtail does not collect data from the log file.
Non-standard text logs	If a log contains \0, the log is truncated at the first occurrence of \0.

Limits on checkpoints

Item	Description

Item	Description
Checkpoint timeout period	If a log file is not modified for more than 30 days, the checkpoint of the log file is deleted.
Checkpoint storage policy	Checkpoints are stored at intervals of 15 minutes and at the point in time when Logtail exits. For more information about how to change the values of the related parameters, see Configure the startup parameters of Logtail.
Checkpoint storage path	By default, checkpoints are stored in the `/tmp/logtail_checkpoint` directory. For more information about how to change the values of the related parameters, see Configure the startup parameters of Logtail.

Limits on Logtail configurations

Item	Description
Configuration update	A configuration update requires approximately 30 seconds to take effect.
Dynamic loading of Logtail configurations	Logtail configurations can be dynamically updated. An update of a Logtail configuration does not affect other Logtail configurations.
Number of Logtail configurations	The number of Logtail configurations is unlimited. However, we recommend that you create no more than 100 Logtail configurations for a server.
Multi-tenant isolation	Logtail configurations are isolated.

Limits on resources and performance metrics

Item	Description
Throughput for log processing	The default transmission speed of raw logs is limited to 20 MB/s. Log data is uploaded after it is encoded and compressed. The compression ratio ranges from 5:1 to 10:1. If the speed exceeds the limit, log data may be lost. For more information about how to change the values of the related parameters, see Configure the startup parameters of Logtail.
Maximum processing speed for logs	Single-core processing speed: The maximum processing speed is 100 MB/s in simple mode, 40 MB/s in delimiter mode, and 30 MB/s in JSON mode. By default, the maximum processing speed is 20 MB/s in full regex mode. The maximum processing speed in full regex mode varies based on the complexity of regular expressions. If multiple processing threads are started, the performance can be improved by 150% to 300%.
Number of monitored directories	Logtail limits the depth of monitored directories to reduce the consumption of user resources. If the upper limit is reached, Logtail stops monitoring additional directories or log files. Logtail can monitor a maximum of 3,000 directories, including subdirectories.

Item	Description
Number of monitored files	By default, a Logtail configuration on each server can be used to monitor a maximum of 10,000 files. The Logtail on each server can monitor a maximum of 100,000 files. Excessive files are not monitored. If the upper limit is reached, you can perform the following operations: • Use more exact names to specify the monitored directories in each Logtail configuration. • Increase the value of the mem_usage_limit parameter to raise the threshold of available Logtail memory resources. For more information, see Configure the startup parameters of Logtail. You can raise the threshold to no more than 2 GB. This way, the maximum number of files that can be monitored by using each Logtail configuration is increased to 100,000, and the maximum number of files that the Logtail on each server can monitor is increased to 1,000,000.
Default resources	By default, Logtail can occupy up to 40% of the CPU and 256 MB of memory. If logs are generated at a high speed, you can change the values of the related parameters. For more information, see Configure the startup parameters of Logtail.
Policy used to process excessive resource consumption	If the amount of resources occupied by Logtail remains higher than the upper limit for more than 5 minutes, Logtail is forcibly restarted. The restart may cause data loss or duplication.

Limits on troubleshooting

Item	Description
Network error handling	If a network error occurs, Logtail automatically retries the data collection task and adjusts the retry interval.
Processing of threshold-crossing events	If a data transmission speed exceeds the upper limit of a Logstore, Logtail blocks log collection and automatically retries the data collection task.
Maximum retry period before timeout	If data fails to be transmitted and the issue lasts for more than 6 hours, Logtail discards the data.
Status self-check	If an exception occurs, Logtail restarts. For example, if an application unexpectedly exits or the resource usage exceeds the specified upper limit, Logtail restarts.

Other limits

Item	Description

Item	Description
Log collection latency	In most cases, a latency of less than 1 second exists between the point in time at which a log is written to disk and the point in time at which Logtail collects the log. However, if the log collection is blocked, the latency increases.
Log upload policy	Before Logtail uploads logs, Logtail aggregates the logs in the same file. Logtail starts to upload logs when the number of logs exceeds 2,000, the total size of logs exceeds 2 MB, or the log collection duration exceeds 3 seconds.

3.11. FAQ about Logtail

This topic provides answers to some frequently asked questions about Logtail.

Category		References
Terms		• FAQ about Logtail • What are the differences among log collection agents?
Heartbeat detection		What do I do if a Logtail machine group has no heartbeats?
Data collection	All scenarios	• Query local collection status • How do I view Logtail collection errors? • What do I do if errors occur when I use Logtail to collect logs? • How do I troubleshoot the common errors that occur when Log Service collects logs? • What do I do if I want to use multiple Logtail configurations to collect logs from a log file?
	Container scenarios	Troubleshoot log collection exceptions in containers
Deployment and control	All scenarios	• What do I do if the IP address is empty in the app_info.json file of Logtail? • How do I update a Logtail configuration after I switch the network type of an ECS instance from the classic network to a VPC? • How do I collect logs from servers in a corporate intranet? • How do I use the automatic diagnostic tool of Logtail?
	Windows server scenarios	What do I do if error messages appear after I install Logtail on a Windows ECS instance?
	Container scenarios	How do I collect container logs from Kubernetes clusters?

Category		References
Log formats and parsing		• How do I modify a regular expression? • How do I optimize regular expressions? • How do I collect different types of logs in full regex mode?

4.Cloud product collection

4.1. Alibaba Cloud service logs

Log Service can collect logs from multiple types of Alibaba Cloud services, such as elastic computing, storage, security, and database services. The logs record operational statistics, such as user operations, running statuses, and business dynamics of Alibaba Cloud services.

> **Note** Log Service uses the Log Audit Service application to collect logs of the following Alibaba Cloud services across accounts: 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. For more information, see Log Audit Service.

The following tables list the Alibaba Cloud services from which Log Service can collect logs. The tables also list the related projects, Logstores, and dashboards that can be created in Log Service.

- Elastic computing

Alibaba Cloud service	Log	Project and Logstore	Dashboard
Elastic Compute Service (ECS)	Full log	Custom project and Logstore Logs are collected by using Logtail. For more information, see Logtail overview.	Custom dashboard
ACK	◦ Kubernetes audit log ◦ Kubernetes event center ◦ Ingress log	Custom project and Logstore Logs are collected by using Logtail. For more information, see Container log collection.	Custom dashboard
Function Compute	Execution log	◦ Project: aliyun-fc-*regionID*-bb47c1e4-cdd4-5318-978e-d748952652c8 ◦ Logstore: function-log	Custom dashboard

- Storage

Alibaba Cloud service	Log	Project and Logstore	Dashboard

Alibaba Cloud service	Log	Project and Logstore	Dashboard
OSS	Access log	○ Project: oss-log-*Alibaba Cloud account ID-region ID* ○ Logstore: oss-log-store	○ Access Center ○ Audit Center ○ Operation Center ○ Performance Center
NAS	Access log	○ Project: nas-*Alibaba Cloud account ID-region ID* ○ Logstore: nas-nfs	○ nas-nfs-nas_summary_dashboard_cn ○ nas-nfs-nas_audit_dashboard_cn ○ nas-nfs-nas_detail_dashboard_cn

- Security

Alibaba Cloud service	Log	Project and Logstore	Dashboard
Anti-DDoS Pro (old version)	Full log	○ Anti-DDoS Pro (old) instances in mainland China ■ Project: ddos-pro-project-*Alibaba Cloud account ID*-cn-hangzhou ■ Logstore: ddos-pro-logstore ○ Anti-DDoS Pro (old) instances outside mainland China ■ Project: ddos-pro-*Alibaba Cloud account ID*-ap-southeast-1 ■ Logstore: ddos-pro-logstore	○ DDoS Operation Center ○ DDoS Access Center

Alibaba Cloud service	Log	Project and Logstore	Dashboard
Anti-DDoS Pro and Anti-DDoS Premium	Full log	◦ Anti-DDoS Pro ■ Project: ddoscoo-project-*Alibaba Cloud account ID*-cn-hangzhou ■ Logstore: ddoscoo-logstore ◦ Anti-DDoS Premium ■ Project: ddosdip-project-*Alibaba Cloud account ID*-ap-southeast-1 ■ Logstore: ddoscoo-logstore	◦ DDoS Operation Center ◦ DDoS Access Center

Alibaba Cloud service	Log	Project and Logstore	Dashboard
SAS	○ Security log ■ Vulnerability log ■ Baseline log ■ Security alert log ○ Network log ■ DNS log ■ Local DNS log ■ Network session log ■ Web access log ○ Host log ■ Process startup log ■ Network connection log ■ Logon log ■ Brute-force attack log ■ Process snapshot log ■ Account snapshot log ■ Port snapshot log	○ Project: sas-log-*Alibaba Cloud ID-region ID* ○ Logstore: sas-log	○ Network log ■ DNS Access Center ■ Network Session Center ■ Web Access Center ○ Host log ■ Login Center ■ Process Center ■ Connection Center ○ Security log ■ Baseline Center ■ Vulnerability Center ■ Alert Center

Alibaba Cloud service	Log	Project and Logstore	Dashboard
WAF	○ Access log ○ Attack log	○ WAF instances in mainland China ■ Project: waf-project-*Alibaba Cloud account ID*-cn-hangzhou ■ Logstore: waf-logstore ○ WAF instances outside mainland China ■ Project: waf-project-*Alibaba Cloud account ID*-ap-southeast-1 ■ Logstore: waf-logstore	○ Operation Center ○ Access Center ○ Security Center
CFW	Internet access log	○ Project: cloudfirewall-project-*Alibaba Cloud account ID*-cn-hangzhou ○ Logstore: cloudfirewall-logstore	Report

- Networking

Alibaba Cloud service	Log	Project and Logstore	Dashboard
SLB	Layer-7 access log	Custom project and Logstore	○ slb-user-log-slb_layer7_operation_center_cn ○ slb-user-log-slb_layer7_access_center_cn
Virtual Private Cloud (VPC)	Flow log	Custom project and Logstore	○ Logstore Name-vpc_flow_log_traffic_cn ○ Logstore Name-vpc_flow_log_rejection_cn ○ Logstore Name-vpc_flow_log_overview_cn

Alibaba Cloud service	Log	Project and Logstore	Dashboard
Elastic IP Address (EIP)	Internet access log	Custom project and Logstore	eip_monitoring
API Gateway	Access log	Custom project and Logstore	Logstore Name_apigateway_access_log

- Database

Alibaba Cloud service	Log	Project and Logstore	Dashboard
ApsaraDB for RDS	SQL audit log	Custom project and Logstore	◦ RDS Audit Operation Center ◦ RDS Audit Performance Center ◦ RDS Audit Security Center
ApsaraDB for Redis	◦ Audit log ◦ Slow query log ◦ Operational log	◦ Project: nosql-*Alibaba Cloud account ID-region ID* ◦ Logstore: ▪ redis_audit_log ▪ redis_slow_run_log	◦ Redis Audit Center ◦ Redis Slow Log Center
ApsaraDB for MongoDB	◦ Audit log ◦ Slow query log ◦ Operational log	◦ Project: nosql-*Alibaba Cloud account ID-region ID* ◦ Logstore: ▪ mongo_audit_log ▪ mongo_slow_run_log	Mongo Audit Log Center

- Management and monitoring

Alibaba Cloud service	Log	Project and Logstore	Dashboard
ActionTrail	ActionTrail access log	◦ Project: custom project ◦ Logstore: actiontrail_trail name	actiontrail_trail name_audit_center_cn
	Operations log	Custom project and Logstore	innertrail_trail name_audit_center_cn

- Internet of Things (IoT)

Alibaba Cloud service	Log	Project and Logstore	Dashboard
IoT Platform	IoT Platform log	○ Project: iot-log-*Alibaba Cloud account ID-region ID* ○ Logstore: iot_logs	IoT Operation Center

4.2. Common operations on logs of Alibaba Cloud services

After you enable the log analysis feature in the console of an Alibaba Cloud service, Log Service collects logs of the Alibaba Cloud service. This topic describes the common operations that you can perform on logs of Alibaba Cloud services in the Log Service console.

Common operations

The following table describes the operations that you can perform after you collect logs of Alibaba Cloud services.

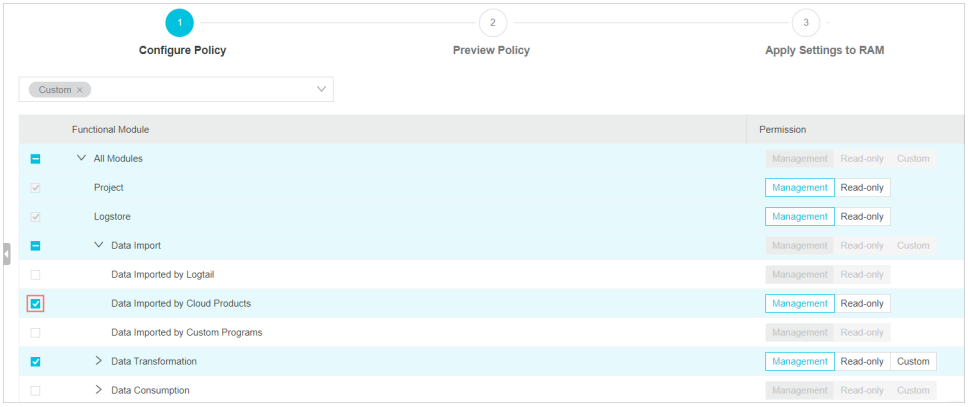
Operation	Description
Query and analyze logs	Query and analyze logs by using query statements. For more information, see Query and analyze logs.
Query and analyze time series data	Query and analyze time series data by using query statements. For more information, see Query and analyze time series data.
View raw logs	View raw logs on the Search & Analysis page.
Perform quick analysis	Perform quick analysis on log fields. For more information, see Quick analysis.
Render results into charts	Render all query results that match the specified query statements into charts. For more information, see Analysis graph
Configure alerts	Create an alert rule for the query results. For more information, see Create an alert rule.
Create a dashboard	Create a dashboard and add an analysis chart to the dashboard. For more information, see Add charts to a dashboard.
Download logs	Download logs to a local computer. For more information, see 下载日志.
Transform data	Standardize, enrich, distribute, and aggregate the collected logs. For more information, see Data transformation.
Consume data	Consume the collected logs. For more information, see Consume logs.

Operation	Description
Ship data	Ship the collected logs to storage or computing services such as Object Storage Service (OSS), MaxCompute, and E-MapReduce (EMR). For more information, see Ship logs.

RAM user authorization

If you need to use a RAM user to manage logs of Alibaba Cloud services, you must use your Alibaba Cloud account to authorize the RAM user.

You can use the permission assistant feature to grant permissions to a RAM user. For more information, see Configure the permission assistant feature.



4.3. Function Compute execution logs

4.3.1. Overview

The log query feature of Function Compute collects the execution logs of functions to Log Service. This feature allows you to debug codes, troubleshoot errors, and analyze data. This topic describes the resources and limits of the log query feature in Function Compute.

Resources

- Dedicated project and Logstore

  After you enable the log query feature of Function Compute, Log Service creates a project named in the format of aliyun-fc-region ID-bb47c1e4-cdd4-5318-978e-d748952652c8 and a Logstore named function-log.

- Dashboard

  After you enable the log query feature of Function Compute, Log Service does not generate a dashboard. You can create a custom dashboard.

Billing

- You are not charged for using the log query feature of Function Compute.
- After function execution logs are collected by Log Service, you are charged for the storage space that the logs occupy, data reads, read/write requests, data transformation, and data shipping. For more information, see Log Service pricing.

Limits

You can write only function execution logs to a dedicated Logstore.

4.3.2. Enable the logging feature

Before you can query and analyze the execution logs of functions, you must enable the logging feature of Function Compute. This topic describes how to enable the logging feature in the Function Compute console.

Prerequisites

Function Compute is activated. For more information, see Activate Function Compute.

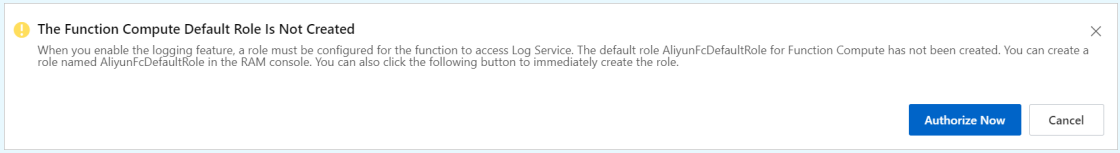
Procedure

> **Note** Before you can use a RAM user to enable the logging feature, you must grant the required permissions to the RAM user. For more information, see RAM user authorization.

1. Log on to the Function Compute console.
2. In the left-side navigation pane, click **Services & Functions**.
3. In the top navigation bar, select a region based on your business requirements.
4. On the **Services** page, click **Create Service**.
5. In the **Create Service** panel, set the parameters and click **OK**.

   The following table describes the parameters. For more information, see Create a service.

   > **Notice** The first time you enable the logging feature, a message appears after you click **OK** in the Create Service panel. You must authorize Function Compute to assume the AliyunFcDefaultRole role. This way, Function Compute can access Log Service. In the message, click **Authorize Now** and complete the authorization as prompted.

Parameter	Description
Service Name	The name of the service.
Logging	Specifies whether to enable the logging feature. ○ **Enable**: If you select Enable, Log Service generates a project named aliyun-fc-region ID-**** and a Logstore named function-log in the corresponding region. Indexing is enabled for the Logstore. ○ **Disable**: If you select Disable, the execution logs of Function Compute are not delivered to Log Service.

What's next

After Log Service collects execution logs, you can query, analyze, download, ship, and transform the logs in the Log Service console. You can also create alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.3.3. Log fields

This topic describes the fields of Function Compute execution log entries.

Log field	Description
__topic__	The topic of a log entry. The value of this field is the same as that of the serviceName field.
functionName	The name of a function.
message	The message of a log entry.
qualifiter	The alias of a service version.
serviceName	The name of a service.
versionId	The ID of a service version.

4.4. OSS access logs

4.4.1. Usage notes

Alibaba Cloud Object Storage Service (OSS) provides the real-time log query feature that is supported by Log Service. You can use this feature to audit operations performed on OSS, analyze access requests, track anomaly events, and locate errors and exceptions. This topic describes the resources, billings, and limits that are related to the real-time log query feature.

Resources

- Dedicated project and Logstore

  If you enable the log query feature of OSS, a project named in the format of oss-Alibaba Cloud account ID-region ID and a Logstore named oss-log-store are created.

- Dedicated dashboards

  After you enable the feature, four dedicated dashboards are automatically created for OSS access logs.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize the results of log queries. For more information, see Create a dashboard.

Dashboard	Description

Dashboard	Description
oss-log-store_access_center	Displays the overall operational statistics of OSS, including the page views (PVs), (unique views) UVs, traffic, and distribution of requests over the external network.
oss-log-store_audit_center	Displays the statistics of operations performed on OSS objects, including the read, write, and delete operations.
oss-log-store_operation_center	Displays the information of OSS O&M operations, including the number of requests and distribution of failed operations.
oss-log-store_performance_center	Displays the statistics of OSS performance, including the performance of downloads and uploads over the external network, transmission performance over different networks or of different object sizes, and list of differences between object downloads.

Billing

- You can use the real-time log query feature to query OSS logs that are generated in the last 7 days free of charge. You can also write a maximum of 900 GB of OSS logs to a dedicated Logstore free of charge. If the size of a log entry is 1 KB, you can write 900 million log entries to a dedicated Logstore. Excess data is charged based on the billing methods of Log Service. If you set a log retention period longer than 7 days, you are charged for the data storage in the excess days based on the billing methods of Log Service. For more information, see Billable items.
- You can use a maximum of 16 shards in a dedicated Logstore to store and query OSS logs free of charge. Excess shards are charged based on the billing methods of Log Service.
- You are charged for reading data from a dedicated Logstore over the external network. You are also charged for transforming and shipping data. For more information, see Pay-as-you-go.

Limits

- You can write only OSS access logs to a dedicated Logstore. In addition, you cannot modify the indexes in a dedicated Logstore.
- A dedicated Logstore cannot be deleted.
- If you have an overdue payment, the real-time log query feature is unavailable.

4.4.2. Enable the real-time log query feature

Before you query Object Storage Service (OSS) access logs in the OSS console, you must enable the real-time log query feature in the OSS console. This topic describes how to enable the real-time log query feature in the OSS console.

Prerequisites

- You have created an Object Storage Service (OSS) bucket. For more information, see Create buckets.

  The Log Service project that is used to store OSS access logs and the OSS bucket belong to the same Alibaba Cloud account and reside in the same region.
- Log Service is authorized to assume the AliyunLogImportOSSRole role to access OSS.

  You can go to the Cloud Resource Access Authorization page to authorize Log Service to access OSS.

> **Note**
> - You need to perform this operation only if Log Service was not authorized to assume the AliyunActionTrailDefaultRole role by using an Alibaba Cloud account.
> - If you use a RAM user to log on to OSS, you must authorize the RAM user by using an Alibaba Cloud account. For more information, see RAM user authorization.
> - You must not delete the RAM role or revoke the permissions from the RAM role. Otherwise, logs cannot be shipped to Log Service.

Procedure

1. Log on to the OSS console.
2. In the left-side navigation pane, click **Buckets**. On the Buckets page, click the name of the bucket for which you want to enable the log query feature.
3. Choose **Logging > Real-time Log Query**.
4. Click **Activate Now**.

   After you enable the real-time log query feature, Log Service immediately collects logs from OSS. At the same time, a dedicated project and a Logstore are automatically created and indexes are created in the Logstore.

What's next

After OSS access logs are collected by Log Service, you can query, download, ship, and transform the logs. You can also create alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.4.3. Log fields

This topic describes different types of Object Storage Service (OSS) logs and the fields in each type of log.

OSS log types

Log type	Description
Access logs	This type of log records access to OSS buckets. Access logs are collected in real time.
Batch deletion logs	This type of log records information about deleted objects. Batch deletion logs are collected in real time. **Note** When you call the DeleteObjects operation, an access log is generated to record the request.
Hourly metering logs	This type of log records hourly metering statistics for an OSS bucket, which helps you analyze data. Hourly metering logs are collected several hours after they are generated.

OSS built-in logging and Log Service-powered OSS logging

OSS provides the real-time log query feature, which is powered by Log Service. If you enable the feature, Log Service collects and stores OSS access logs, batch deletion logs, and hourly metering logs of OSS. You can perform various operations on the logs. For example, you can query and analyze the logs. OSS built-in logging is a built-in feature of OSS. This feature records and stores the access logs of OSS.

Log Service records the same access logs as OSS, but the log fields are different. For more information, see Access logs.

Field of OSS access logs recorded in OSS	Field of OSS access logs recorded in Log Service
Remote IP	client_ip
Time	time
Request-URI	request-uri
HTTP Status	http_status
SentBytes	response_body_length
RequestTime (ms)	response_time
Referer	referer
User-Agent	user-agent
HostName	host
Request ID	request_id
LoggingFlag	logging_flag
Requester Aliyun ID	requester_id
Operation	operation
Bucket	bucket
Key	object
ObjectSize	object_size
Server Cost Time (ms)	server_cost_time
Error Code	error_code
Request Length	request_length
UserID	owner_id
Delta DataSize	delta_data_size
Sync Request	sync_request

Access logs

Field	Description
__topic__	The topic of the log. The value is fixed as oss_access_log.
acc_access_region	The region of the OSS endpoint that is specified in the request. If the request is sent when the transfer acceleration feature is enabled, the value of this field is the name of the region. If the request is sent when the transfer acceleration feature is disabled, the value of this field is a hyphen (-).
access_id	The AccessKey ID of the requester.
bucket	The name of the OSS bucket.
bucket_location	The data center where the OSS bucket resides. The value is in the oss-<Region ID> format.
bucket_storage_type	The storage class of the OSS Object. Valid values: • standard: the Standard storage class • archive: the Archive storage class • infrequent_access: the Infrequent Access storage class
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.
content_length_in	The value of the Content-Length header in the request. Unit: bytes.
content_length_out	The value of the Content-Length header in the response. Unit: bytes.
delta_data_size	The change to the size of an OSS object. If the object size does not change, the value of this field is 0. If the request is not an upload request, the value of this field is a hyphen (-).
error_code	The error code that is returned by OSS. For more information, see Error responses.
host	The endpoint of the OSS bucket. Example: bucket123.oss-cn-beijing.aliyuncs.com.
http_method	The HTTP request method.
http_status	The HTTP status code.
http_type	The protocol over which the request is sent. Valid values: http and https.
logging_flag	Indicates whether logging is enabled to export logs to an OSS bucket at regular intervals. A value of true indicates that the feature is enabled.
object	The URL-encoded object that is requested. You can include `select url_decode(object)` in your query statement to decode the object.

Field	Description
object_size	The size of the object. Unit: bytes.
operation	The API operation. For more information, see API operations.
owner_id	The ID of the Alibaba Cloud account to which the OSS bucket belongs.
referer	The Referer header of the HTTP request.
request_id	The ID of the request.
request_length	The size of the HTTP request. The headers of the request are counted. Unit: bytes.
request_uri	The URL-encoded URI of the HTTP request. The query string is included. You can include `select url_decode(request_uri)` in your query statement to decode the URI.
requester_id	The ID of the requester. If the request is sent by an anonymous user, the value of this field is a hyphen (-).
response_body_length	The size of the HTTP response body.
response_time	The HTTP response time. Unit: milliseconds.
server_cost_time	The time that is consumed by the OSS server to process the request. Unit: milliseconds.
sign_type	The type of the signature. Valid values: • NotSign: The request is not signed. • NormalSign: The request is signed by using a regular signature. • UriSign: The request is signed by using a URL signature. • AdminSign: The request is signed by using an administrator account.
sync_request	The type of the synchronous request. Valid values: • hyphen (-): general request • cdn: CDN back-to-origin request • sync-public: cross-region replication request • lifecycle: lifecycle rule configuration request
time	The time at which OSS receives the request. Example: 27/Feb/2018:13:58:45. If you need a timestamp, you can use the __time__ field.
user-agent	The User-Agent header of the HTTP request. Example: curl/7.15.5.
vpc_addr	The IP address of the server that resides in a VPC and hosts the OSS bucket.
vpc_id	The ID of the VPC.

Field	Description
delta_data_size	The change to the size of an OSS object. If the object size does not change, the value of this field is 0. If the request is not an upload request, the value of this field is a hyphen (-).
acc_access_region	The region of the OSS endpoint that is specified in the request. If the request is sent when the transfer acceleration is enabled, the value of this field is the ID of the region. If the request is sent when the transfer acceleration is disabled, the value of this field is a hyphen (-).
restore_priority	The priority of the log in the event of log data restoration.

Batch deletion logs

When you call the DeleteObjects operation, an access log is generated to record the request. 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 is a hyphen (-). If you want to view the information about deleted objects, you must obtain batch deletion logs. You can use the request_id field in a batch deletion log to find the request that is used to delete objects. The following table describes the fields in a batch deletion log.

Field	Description
__topic__	The topic of the log. The value is fixed as oss_batch_delete_log.
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 header of the HTTP request. Example: curl/7.15.5.
bucket	The name of the OSS bucket.
error_code	The error code that is returned by OSS. For more information, see Error responses.
request_length	The size of the HTTP request. The headers of the request are counted. Unit: bytes.
response_body_length	The size of the HTTP response body.
object	The URL-encoded object that is requested. You can include `select url_decode(object)` in your query statement to decode the object.
object_size	The size of the object. Unit: bytes.
operation	The API operation. For more information, see API operations.
bucket_location	The data center where the OSS bucket resides. The value is in the oss-<Region ID> format.
http_method	The HTTP request method. Example: POST.

Field	Description
referer	The Referer header of the HTTP request.
request_id	The ID of the request.
http_status	The HTTP status code.
sync_request	The type of the synchronous request. Valid values: • hyphen (-): general request • cdn: CDN back-to-origin request • sync-public: cross-region replication request
request_uri	The URL-encoded URI of the request. The query string is included. You can include `select url_decode(request_uri)` in your query statement to decode the URI.
host	The endpoint of the OSS bucket. Example: bucket123.oss-cn-beijing.aliyuncs.com.
logging_flag	Indicates whether logging is enabled to export logs to an OSS bucket at regular intervals. A value of true indicates that the feature is enabled.
server_cost_time	The time that is consumed by the OSS server to process the request. Unit: milliseconds.
owner_id	The ID of the Alibaba Cloud account to which the OSS bucket belongs.
requester_id	The ID of the requester. If the request is sent by an anonymous user, the value of this field is a hyphen (-).
delta_data_size	The change to the size of an OSS object. If the object size does not change, the value of this field is 0. If the request is not an upload request, the value of this field is a hyphen (-).

Hourly metering logs

Hourly metering logs record hourly metering statistics for an OSS bucket, which helps you analyze data.

Field	Description
__topic__	The topic of the log. The value is fixed as oss_metering_log.
owner_id	The ID of the Alibaba Cloud account to which the OSS bucket belongs.
bucket	The name of the OSS bucket.
cdn_in	The inbound traffic from CDN to the OSS bucket. Unit: bytes.
cdn_out	The outbound traffic from the OSS bucket to CDN. Unit: bytes.
get_request	The number of GET requests.

Field	Description
intranet_in	The inbound traffic over an internal network. Unit: bytes.
intranet_out	The outbound traffic over an internal network. Unit: bytes.
network_in	The inbound traffic over the Internet. Unit: bytes.
network_out	The outbound traffic over the Internet. Unit: bytes.
put_request	The number of PUT requests.
storage_type	The storage class of the OSS bucket. Valid values: • standard: the Standard storage class • archive: the Archive storage class • infrequent_access: the Infrequent Access storage class
storage	The amount of data in the OSS bucket. Unit: bytes.
metering_datasize	The amount of data whose storage class is not Standard in the OSS bucket.
process_img_size	The size of the image that is processed. Unit: bytes.
process_img	The information about the image that is processed.
sync_in	The inbound traffic that is generated by the synchronization operation. Unit: bytes.
sync_out	The outbound traffic that is generated by the synchronization operation. Unit: bytes.
start_time	The start timestamp of a metering operation.
end_time	The end timestamp of a metering operation.
region	The region where the OSS bucket resides.
bucket_location	The data center where the OSS bucket resides. The value is in the oss-<Region ID> format.

API operations

The following table describes the supported API operations. For more information, see List of operations by function.

Operation	Description
AbortMultiPartUpload	Cancels a multipart upload task.
AppendObject	Appends an object to an existing object.
CompleteUploadPart	Completes the multipart upload tasks of an object.

Operation	Description
CopyObject	Copies an object.
DeleteBucket	Deletes a bucket.
DeleteLiveChannel	Deletes a LiveChannel.
DeleteObject	Deletes an object.
DeleteObjects	Deletes multiple objects.
GetBucket	Queries the information about 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 that are configured for the objects in a bucket.
GetBucketLocation	Queries the region where a bucket resides.
GetBucketLog	Queries the access logging configurations of a bucket.
GetBucketReferer	Queries the hotlink protection rules that are configured for a bucket.
GetBucketReplication	Queries the cross-region replication (CRR) rules that are configured for a bucket.
GetBucketReplicationProgress	Queries the progress of a CRR task that is performed on a bucket.
GetBucketStat	Queries the information about a bucket.
GetBucketWebSite	Queries the status of static website hosting for a bucket.
GetLiveChannelStat	Queries the status of a LiveChannel.
GetObject	Queries an object.
GetObjectAcl	Queries the ACL of an object.
GetObjectInfo	Queries the information about an object.

Operation	Description
GetObjectMeta	Queries the metadata of an object.
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.
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 tasks in progress.
ListParts	Queries all the parts that are uploaded in a specified multipart upload task.
PostObject	Uploads an object by using an HTML form.
PostProcessTask	Commits data processing operations, such as screenshot.
PostVodPlaylist	Creates a playlist for a LiveChannel.
ProcessImage	Processes an image.
PutBucket	Creates a bucket.
PutBucketCors	Configures CORS rule for a bucket.
PutBucketLifecycle	Configures lifecycle rules for a bucket.
PutBucketLog	Configures access logging for a bucket.
PutBucketWebSite	Configures the static website hosting mode for a bucket.
PutLiveChannel	Creates a LiveChannel.
PutLiveChannelStatus	Configures the status of a LiveChannel.
PutObject	Uploads an object.
PutObjectAcl	Modifies the ACL of an object.

Operation	Description
PutObjectSymlink	Creates a symbolic link for an object.
RedirectBucket	Redirects a request to a bucket endpoint.
RestoreObject	Restores an object.
UploadPart	Uploads an object by part.
UploadPartCopy	Copies data from an existing object to upload a part.
get_image_exif	Queries the 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 a style of a bucket.
list_style	Queries all styles of a bucket.
put_style	Creates a style for a bucket.

4.5. NAS access logs

4.5.1. Usage notes

The log analysis feature of Apsara File Storage NAS allows you to query, analyze, transform, and consume logs that are generated when you access NAS data. This topic describes the assets, billing, and limits of the log analysis feature in NAS.

Assets

- Dedicated projects and dedicated Logstores

  After you enable the log analysis feature of NAS, Log Service creates a project named nas-Alibaba Cloud account ID-region ID and a Logstore named nas-nfs in each region.

  > **Note** To ensure that logs can be collected, do not delete dedicated projects and dedicated Logstores.

- Dedicated dashboards

  After you enable the log analysis feature of NAS, Log Service generates three dashboards by default.

> **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize the results of log analysis. For more information, see Create a dashboard.

Dashboard	Description
nas-nfs-nas_summary_dashboard_cn	Shows the operational statistics of NAS. The information displayed on the dashboard includes the number of the latest accessed volumes, the number of the latest unique visitors (UVs), total write throughput, and total read throughput.
nas-nfs-nas_audit_dashboard_cn	Shows the operational statistics of NAS file systems. The information displayed on the dashboard includes the number of create operations, deleted files, and read files.
nas-nfs-nas_detail_dashboard_cn	Shows the details of NAS file systems. The information displayed on the dashboard includes the number of the latest page views (PVs) and operation trend.

Billing

- You are not billed for the log analysis feature in NAS.
- After NAS access logs are dumped to Log Service, you are billed based on the storage space, read traffic, the number of requests, data transformation, and data shipping in the Log Service console. For more information, see Log Service pricing.

Limits

- You can write only NAS access log data to a dedicated Logstore.
- The log analysis feature supports only a file system that uses the Network File System (NFS) protocol.

4.5.2. Enable the log analysis feature

This topic describes how to enable the log analysis feature in the Apsara File Storage NAS console. After you enable the log analysis feature, you can dump NAS access logs to Log Service.

Prerequisites

- A file system that uses the Network File System (NFS) protocol is created and mounted on a server. For more information, see Mount an NFS file system on a Linux ECS instance.
- NAS is authorized to use the AliyunNASLogArchiveRole role to access Log Service.

  You can go to the Cloud Resource Access Authorization page to complete the authorization.

> **Note**
> - This operation is required only when you enable the log analysis feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
> - If you use a RAM user to log on to NAS, you must grant required permissions to the RAM user. For more information, see RAM user authorization.
> - To ensure that NAS access logs can be dumped to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

Procedure

1. Log on to the NAS console.
2. In the left-side navigation pane, choose **Monitoring Audit > Log Analysis**.
3. On the **Log Analysis** page, click **New Log Dump**.
4. In the **New Log Dump** dialog box, select a system type from the **File System Type** drop-down list, select a file system from the **File System ID/Name** drop-down list, and then click **OK**.

What's next

After NAS access logs are dumped to Log Service, you can query, analyze, download, ship, and transform the dumped logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.5.3. Log fields

This topic describes the fields of NAS access log entries.

Log field	Description
__topic__	The topic of a log entry. Valid value: nas_audit_log.
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.
OpList	The procedure number of the NFSv4 protocol.
Proc	The procedure number of the NFSv3 protocol.
RWSize	The size of read/write traffic. 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.
User	The ID of an Alibaba Cloud account.

Log field	Description
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.

4.6. Anti-DDoS Pro logs

4.6.1. Usage notes

The full log feature of Anti-DDoS Pro and Anti-DDoS Premium of the previous version can be used to collect, query, analyze, transform, and consume website access logs and HTTP flood attack logs in real time. You can use this feature to fix website access errors, trace HTTP flood attacks, and analyze website operations. This topic describes the assets, billing, and limits of the full log feature.

> **Note** Anti-DDoS Pro and Anti-DDoS Premium of the previous version has been upgraded to Anti-DDoS Pro. The console of Anti-DDoS Pro and Anti-DDoS Premium of the previous version is available only if you have purchased Anti-DDoS Pro and Anti-DDoS Premium instances. You can no longer create Anti-DDoS Pro and Anti-DDoS Premium instances in the console of Anti-DDoS Pro and Anti-DDoS Premium of the previous version.

Benefits

- Ease of use: To enable the full log feature, you only need to perform a few simple operations. The feature ensures that Anti-DDoS Pro and Anti-DDoS Premium logs are collected in real time. After you add a website to an Anti-DDoS Pro and Anti-DDoS Premium instance, the logs of the website are automatically collected.
- Real-time analysis: The full log feature provides real-time log analysis and out-of-the-box dashboards by using Log Service. The dashboards provide insights into the HTTP flood attacks and access details.
- Real-time monitoring: You can use Log Service to monitor specified metrics and send alerts in real time. This allows you to handle business exceptions at the earliest opportunity.
- High compatibility: Log Service is compatible with other big data solutions, such as stream processing, cloud storage, and visualization. This allows you to extract more value from your business data.

Assets

- A dedicated project and dedicated Logstore
  - After you enable the full log feature for an Anti-DDoS Pro and Anti-DDoS Premium of the previous version instance deployed in mainland China, Log Service creates a project named DDoS-pro-project-Alibaba Cloud account ID-cn-hangzhou. In addition. Log Service creates a Logstore named DDoS-pro-logstore.

- After you enable the full log feature for an Anti-DDoS Pro and Anti-DDoS Premium of the previous version instance deployed outside mainland China, Log Service creates a project named DDoS-pro-Alibaba Cloud account ID-ap-southeast-1. In addition, Log Service creates a Logstore named DDoS-pro-logstore.

  > **Note** You can write only the logs of the Anti-DDoS Pro and Anti-DDoS Premium of the previous version instance to the dedicated Logstore. However, this limit does not apply to the logs that are related to statistics, alerts, and consumption.

- Dedicated dashboards

  By default, Log Service generates two dedicated dashboards for the logs of the Anti-DDoS Pro or Premium of the previous version.

  > **Note** We recommend that you do not make changes to the dedicated dashboard. This may affect the usability of the dashboards. You can create a custom dashboard to view log analysis results. For more information, see Create a dashboard.

Dashboard	Description
DDoS Operation Center	Displays the operational statistics of protected websites. The statistics include the rate of valid requests, amount of valid traffic, number of requests and interceptions, and overview of attacks.
DDoS Access Center	Displays the access statistics of the protected websites. The statistics include page views (PVs), unique visitors (UVs), inbound traffic, peak inbound bandwidth, peak outbound bandwidth, PV/UV trends, and source distribution.

Billing

- Anti-DDoS Pro and Anti-DDoS Premium of the previous version provides the following free quotas. If the number of resources that you have used exceeds the quota, you are charged for the excess based on the standard pricing.
  - 100 GB of write traffic per day
  - Free rental of four Shard per day
  - Three days of free log retention
- AfterAnti-DDoS Pro and Anti-DDoS Premium of the previous version pushes logs to Log Service, you can query, analyze, monitor, and view log analysis results free of charge. However, you are charged based on the standard pricing of Log Service when you read, transform and ship data, or send alerts by using SMS or voice messages. For more information, see Log Service pricing.

4.6.2. Enable the full log feature

This topic describes how to enable the full log feature in the anti-DDoS Pro (old) console. This feature allows you to collect logs of Anti-DDoS Pro and send the logs to Log Service.

Prerequisites

- An Anti-DDoS Pro instance is created. A domain is added to the instance.

Anti-DDoS Pro has been upgraded to Anti-DDoS Pro (BGP). The Anti-DDoS Pro (old) console is available only for users who have purchased Anti-DDoS Pro instances. You can no longer create Anti-DDoS Pro instances in the Anti-DDoS Pro (old) console.

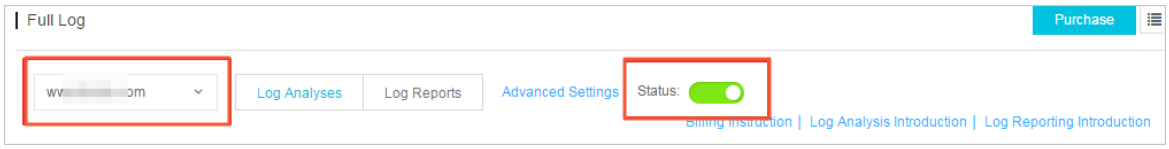
- Log Service is activated.

Procedure

1. Log on to the Anti-DDoS Pro (old) console.
2. In the left-side navigation pane, choose **Log > Full Log**.
3. On the **Full Log** page, authorize Anti-DDoS Pro to use the AliyunDDoSCOOLogArchiveRole role to access Log Service.

   > **Note**
   > - This operation is required only when you enable the full log feature for the first time.
   > - If you use a RAM user to log on to the Anti-DDoS Pro console, you must grant the required permissions to the RAM user.
   > - To ensure that logs can be pushed to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

4. Select the website domain that you want to enable the full log feature, turn on the **Status** switch.



What's next

After the logs of Anti-DDoS Pro are pushed to Log Service, you can query, analyze, download, ship, and transform the collected logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.6.3. Log fields

This topic describes the fields of Anti-DDos Pro log entries.

Log field	Description
__topic__	The topic of a log entry. Valid value: ddos_access_log.
body_bytes_sent	The size of a request body. Unit: bytes.
cache_status	The cache status.
cc_action	The action that is performed to block HTTP flood attacks, for example, challenge, pass, close, captcha, wait, or login. If no action is performed, "none" is displayed.

Log field	Description
cc_phase	The protection policy that is used to block HTTP flood attacks, for example, 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 a protection policy. • If the value is *1*, the request is blocked. • If the value is not 1, the request is allowed.
content_type	The content type of a request.
host	The origin server.
http_cookie	The cookie of a request.
http_referer	The referer of a request. If an HTTP header does not contain a referer, a hyphen (-) is displayed.
http_user_agent	The user agent of a request.
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), for example, BGP, China Telecom, or China Unicom.
matched_host	The matched origin server. This can be a wildcard domain name. If no origin server is matched, a hyphen (-) is displayed.
querystring	The string of a request.
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 for which a request is processed. Unit: microseconds.
request_uri	The uniform resource identifier (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, for example, 200.
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_browser_version	The version 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 that are separated by commas (,). Each IP address is in the IP:Port format.
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.
user_id	The ID of an Alibaba Cloud account.

4.7. Anti-DDoS Pro and Anti-DDoS Premium logs

4.7.1. Usage notes

The log analysis feature of Alibaba Cloud Anti-DDoS Pro and Anti-DDoS Premium allows you to query, analyze, transform, and consume website access logs and HTTP flood attack logs in real time. You can use the log analysis feature to troubleshoot website access errors, trace HTTP flood attacks, and analyze website operations. This topic describes the assets, billing, and limits of the log analysis feature.

Alibaba Cloud provides the following two Anti-DDoS solutions based on the regions where your servers are deployed: Anti-DDoS Pro and Anti-DDoS Premium. In the top navigation bar of the Anti-DDoS Pro console, you can select a region where the servers of Anti-DDoS Pro or Anti-DDoS Premium are deployed. Anti-DDoS Pro applies to scenarios where servers are deployed in mainland China. Anti-DDoS Premium applies to scenarios where servers are deployed outside mainland China.

Assets

- Dedicated projects and dedicated Logstores
  - Anti-DDoS Pro

    After you enable the log analysis feature of Anti-DDoS Pro, Log Service creates a project named ddoscoo-project-Alibaba Cloud account ID-cn-hangzhou and a Logstore named ddoscoo-logstore.
  - Anti-DDoS Premium

    After you enable the log analysis feature of Anti-DDoS Premium, Log Service creates a project named ddosdip-project-Alibaba Cloud account ID-ap-southeast-1 and a Logstore named ddosdis-logstore.
- Dedicated dashboards

  After you enable the log analysis feature of Anti-DDoS Pro or Anti-DDoS Premium, Log Service generates two dashboards by default.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can manually create a dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
DDoS Operation Center	Shows the operational statistics of the websites that are protected by Anti-DDoS Pro or Anti-DDoS Premium.
DDoS Access Center	Shows the access statistics of the websites that are protected by Anti-DDoS Pro or Anti-DDoS Premium.

Billing

You are billed for the log analysis feature when you enable the log analysis feature in the Anti-DDoS Pro or Anti-DDoS Premium console. You are also billed when you transform logs, ship logs, and read data from the Internet in the Log Service console. For more information, see Billable items.

Limits

- You can write only Anti-DDoS log data to a dedicated Logstore.
- If you have overdue payments for your Log Service resources, the log analysis feature is automatically stopped. To ensure service continuity, you must pay your overdue payment within the prescribed time limit.
- You cannot delete a dedicated Logstore or modify its data retention period in the Log Service console. You can modify the data retention period only in the Anti-DDoS Pro or Anti-DDoS Premium console. You can specify a retention period. The retention period must be between 30 and 180 days.
- If the storage space of a dedicated Logstore is full, no more logs can be written to the Logstore.

  > **Note** You can view the usage of log storage space in the Anti-DDoS Pro or Anti-DDoS Premium console. However, the usage is not updated in real time. The storage space usage displayed is delayed for two hours.

- The log analysis feature must be within the validity period. If you do not renew the feature within seven days after expiration, all logs stored in the dedicated Logstore are automatically deleted.

Benefits

- Simple configuration: You can perform a few simple steps to enable the analysis feature and start to collect Anti-DDoS logs in real time. After you add a website to an Anti-DDoS instance, logs of the website are automatically collected.
- Real-time analysis: Based on Log Service, the log analysis feature provides real-time analysis and log reports of HTTP flood attacks and access details.
- Real-time alerts: The log analysis feature can be used to monitor specified indicators and send alerts in real time. This helps you resolve business exceptions at the earliest opportunity.
- High compatibility: The log analysis feature is compatible with other data solutions such as stream processing, cloud storage, and visualization. This allows you to maximize the value of your business data.

4.7.2. Enable the log analysis feature

This topic describes how to enable the log analysis feature in the Anti-DDoS Pro or Anti-DDoS Premium console. The log analysis feature allows you to collect the logs of Anti-DDoS Pro or Anti-DDoS Premium to Log Service.

Prerequisites

- An Anti-DDoS Pro or Anti-DDoS Premium instance is created. For more information, see Purchase an Anti-DDoS Pro or Anti-DDoS Premium instance.
- One or more websites are added to the Anti-DDoS Pro or Anti-DDoS Premium instance. For more information, see Add a website.

Procedure

> **Note** If you use a RAM user to log on to Anti-DDoS Pro or Anti-DDoS Premium, you must grant required permissions to the RAM user. For more information, see RAM user authorization.

1.
2. In the upper-left corner of the page, select **Mainland China**.

   After you select Mainland China, the Anti-DDoS Pro console appears. If you want to use Anti-DDoS Premium, select **Outside Mainland China**.
3. In the left-side navigation pane, choose **Investigation > Log Analysis**.
4. Select a log storage capacity and duration on the buy page.

   If you have purchased a log storage capacity and duration, skip this step.

   i. Click **Purchase Now**.

ii. On the buy page, set the required parameters. The following table describes the parameters.

Parameter	Description
Edition	Select **Anti-DDoS Pro**. In this example, select Anti-DDoS Pro. If you want to use Anti-DDoS Premium, select **Anti-DDoS Premium**.
Log Storage	Select a log storage capacity. Unit: TB. If the log storage space is full, no more logs can be stored. In most cases, each request log occupies about 2 KB of storage space. If the average request volume of your workload is 500 queries per second (QPS), the storage space required for one day is: 500 × 60 × 60 × 24 × 2 = 86,400,000 KB (about 82 GB). By default, logs are stored for 180 days. If you need to store logs that are generated in the last 180 days, you must select a log storage capacity that is larger than 14,832 GB (about 14.5 TB).
Duration	Select a validity period for the log analysis feature. After the validity period expires, no more logs can be stored. **Warning** If the log analysis feature expires and is not renewed within seven days, the Anti-DDoS Pro server clears all logs that are stored in the Logstore of your instance.

iii. Click **Buy Now** to complete the payment.

5. On the **Log Analysis** page, authorize Anti-DDoS Pro to use the AliyunDDoSCOOLogArchiveRole role to access Log Service.

   Note
   - This operation is required only when you enable the log analysis feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
   - To ensure that logs can be collected to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

6. On the **Log Analysis** page, select the website domain, and turn on the Status switch to enable the log analysis feature.

   Note We recommend that you check the remaining log storage space and validity period at regular intervals when you use the log analysis feature.
   - If the storage space usage exceeds 70%, we recommend that you upgrade the log storage space. Otherwise, you cannot store additional logs and you may lose data.
   - If a large amount of storage space remains unused for a long time, you can downgrade the storage capacity as needed.

What's next

After the logs of Anti-DDoS Pro or Anti-DDoS Premium are collected to Log Service, you can query, analyze, download, ship, and transform the collected logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.7.3. Manage log storage

This topic describes how to manage the log storage space in the Anti-DDoS Pro or Anti-DDoS Premium console.

Procedure

1. Log on to the Anti-DDoS Pro console.
2. In the upper-left corner of the page, select **Mainland China**.

   After you select Mainland China, the Anti-DDoS Pro console appears. If you want to use Anti-DDoS Premium, select **Outside Mainland China**.
3. In the left-side navigation pane, choose **Investigation > Log Analysis**.
4. On the **Log Analysis** page, manage the log storage space.

   To manage the log storage space, perform the following operations as required:
   - View the log storage space.

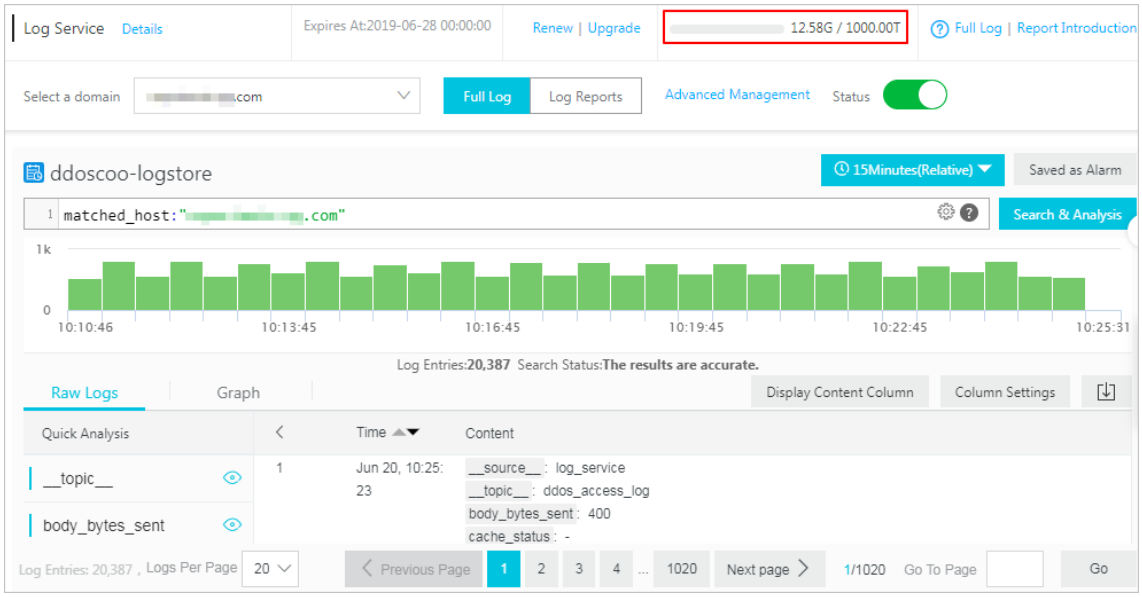
     > **Note** You can view the usage of the log storage space in the Anti-DDoS Pro or Anti-DDoS Premium console. However, the usage is not updated in real time. The storage space usage displayed is delayed by 2 hours. If the storage usage exceeds 70%, you must upgrade the log storage space. Otherwise, you cannot store more logs and data loss may occur.
   - Extend the validity period of the log analysis feature.

     Click **Renew** and select a duration.
   - Upgrade the log storage space.

     Click **Upgrade** to upgrade the storage space.
   - Clear log data.

     Click **Clear** to clear log data.

     > **Warning** You cannot restore log data that is cleared. Proceed with caution.



4.7.4. Log fields

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

Log field	Description
__topic__	The topic of a log entry. • Valid value for Anti-DDoS Pro: ddoscoo_access_log. • Valid value for Anti-DDoS Premium: ddosdip_access_log.
body_bytes_sent	The size of a request body. Unit: bytes.
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.
matched_host	The matched origin server, which can be a wildcard domain name. If no origin server is matched, a hyphen (-) is displayed.

Log field	Description
real_client_ip	The real IP address of a client. If no real IP address can be obtained, a hyphen (-) is displayed.
isp_line	The information of an Internet service provider (ISP) line, for example, BGP, China Telecom, or China Unicom.
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: milliseconds.
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.
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_blocks	Indicates whether a request is blocked by an HTTP flood protection policy. • If the value is *1*, the request is blocked. • If the value is not 1, the request is passed. If this field does not exist, the last_result field is displayed to indicate whether the request is blocked by an HTTP flood protection policy.
last_result	Indicates whether a request is blocked by an HTTP flood protection policy. Valid values: • ok: The request is allowed. • failed: The request is blocked, or the verification fails. If this field does not exist, the cc_blocks field is displayed to indicate whether the request is blocked by an HTTP flood protection policy.
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.
ua_browser	The browser. This field may not exist.

Log field	Description
ua_browser_family	The family to which a browser belongs. This field may not exist.
ua_browser_type	The type of a browser. This field may not exist.
ua_browser_version	The version of a browser. This field may not exist.
ua_device_type	The type of a client. This field may not exist.
ua_os	The operating system of a client. This field may not exist.
ua_os_family	The family of the operating system that runs on a client. This field may not exist.
upstream_addr	The list of back-to-origin IP addresses that are separated by commas (,). Each IP address is in the IP:Port format.
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.
user_id	The ID of an Alibaba Cloud account.
querystring	The string of a request.

4.8. Anti-DDoS Origin logs

4.8.1. Usage notes

Log Service integrates Alibaba Cloud Anti-DDoS Origin to provide the mitigation analysis feature. After you enable the mitigation analysis feature, you can query and analyze mitigation logs that record the events of an Anti-DDoS Origin instance. The events include traffic scrubbing, blackhole filtering, and traffic rerouting. The feature can be used to identify website access exceptions and analyze website operations. This topic describes the assets, billing, and limits of the mitigation analysis feature for Anti-DDoS Origin.

Assets

- A dedicated project and dedicated Logstore

By default, log Service is used to create a project by default named ddosbgp-project-Alibaba Cloud account ID-cn-hangzhou and a Logstore named ddosbgp-logstore. This applies after you enable the log analysis feature of Anti-DDoS Origin.

- Dedicated dashboards

  By default, Log Service generates two dedicated dashboards for the mitigation logs of Anti-DDoS Origin.

  > **Note** We recommend that you do not make changes to the dedicated dashboards. This may affect the usability of the dashboards. You can create a custom dashboard to view log analysis results. For more information, see Create a dashboard.

Dashboards	Description
Anti-DDoS Origin Events Report	The report records Anti-DDoS Origin statistics on blackhole filtering and traffic rerouting for protected websites.
Anti-DDoS Origin Mitigation Report	The report records how Anti-DDoS Origin scrubs the attack traffic of the protected websites. The report includes data such as Inbound Traffic Monitor, Distribution of Inbound Traffic (sort by scrub center) and Protocol of Inbound Traffic.

Billing

- If you enable the log analysis feature in the Anti-DDoS Origin console, you are billed based on the log retention period and storage space. During the public preview analysis step, the full log analysis and event reports of protected traffic for Anti-DDoS Origin is provided free of charge.
- Anti-DDoS Origin pushes logs to Log Service, you can query, analyze, monitor, and view log analysis results free of charge. However, you are charged based on the standard pricing of Log Service when you read, transform and ship data, or send alerts by using SMS or voice messages. For more information, see Log Service pricing.

Limits

- You can write only Anti-DDoS Origin logs to a dedicated Logstore.
- You cannot delete a dedicated Logstore.
- You cannot modify the log retention period for a dedicated Logstore on the Log Service console. However, you can modify the log retention period in the Anti-DDoS Origin console. You can set the value. The value ranges from 30 to 180 days.
- If the storage space of a dedicated Logstore is full, no more logs can be written to the Logstore.

  > **Note** You can view the usage of log storage space in the Anti-DDoS Origin console. However, the usage is not updated in real time. The displayed usage is delayed by two hours.

4.8.2. Enable the mitigation analysis feature of Anti-DDoS Origin

This topic describes how to enable the mitigation analysis feature in the Anti-DDoS Origin console. After you enable the mitigation analysis feature, you can query and analyze Anti-DDoS Origin logs and render query and analysis results into visualized charts in the Log Service console. This helps you protect your business against DDoS attacks.

Prerequisites

An Anti-DDoS Origin instance is created. For more information, see Purchase an Anti-DDoS Origin Enterprise instance.

Procedure

1.
2. In the left-side navigation pane, choose **Network Security > Anti-DDoS Origin > Mitigation Analysis**.
3. In the top navigation bar, select the resource group and region of your instance.
4. The first time you enable the mitigation analysis feature, you must complete RAM authorization as prompted.
5. Upgrade an Anti-DDoS Origin instance.
   i. On the **Mitigation Analysis** page, select the Anti-DDoS Origin instance that you want to upgrade from the **Instance** drop-down list.
   ii. Click **Upgrade Now**.
   iii. On the **Upgrade/Downgrade** page, select On for **Mitigation Analysis (Beta)**.
   iv. Read and select **Anti-DDoS origin Terms of Service**. Then, click **Buy Now**.
   v. Complete the payment.

   > **Note** During public preview, the mitigation analysis feature of Anti-DDoS Origin is provided free of charge.
6. On the **Mitigation Analysis** page, click **Enable Now**.

   After you enable the feature, Anti-DDoS Origin collects mitigation logs stored in Log Service from the instance that is being used. This way, you can query and analyze mitigation logs and view mitigation reports. You can turn on or off Status to enable or disable the feature.

   > **Note** When you use the log analysis feature of Anti-DDoS Origin, we recommend that you check the remaining log storage space and the validity period on a regular basis. If more than 70% of storage space is used, we recommend that you increase the log storage capacity. Otherwise, you cannot store additional logs and data loss may occur.

What's next

After Log Service collects Anti-DDoS Origin logs, you can query, analyze, download, ship and transform the logs. You can also create alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.8.3. Log fields

This topic describes the fields of access logs in Anti-DDoS Origin.

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

4.9. Security Center logs

4.9.1. Usage notes

The log analysis feature of Security Center allows you to collect, query, analyze, transform, and consume risk data in real time. You can use the log analysis feature to monitor and handle potential risks and implement centralized management of cloud resources. This topic describes the assets, billing, and limits of the log analysis feature in Security Center.

Assets

- Dedicated projects and dedicated Logstores

  After you enable the log analysis feature of Security Center, Log Service creates a project named sas-log-Alibaba Cloud account ID-region name and a Logstore named sas-log.

  > **Note** If you accidentally delete the dedicated Logstore, you are prompted that the sas-log Logstore does not exist. All log data in the Logstore is deleted. In this case, you must submit a ticket to reset the log analysis feature. You must re-enable the log analysis feature after it is reset. The deleted data cannot be recovered.

- Dedicated dashboards

  Security Center logs are classified into 3 types and 14 subtypes. After you enable the log analysis feature of Security Center, Log Service generates nine dashboards by default.

Log type	Dashboard	Description
Network log	DNS Access Center	Provides an overview of the DNS queries on the server. The metrics displayed on the dashboard include the success rate of external DNS queries, the distributions of internal and external DNS queries, and the trends of internal and external DNS queries.
	Network Session Center	Provides an overview of resource-related network sessions. The metrics displayed on the dashboard include the trend of network sessions and the distributions of network protocols, source and destination IP addresses, and relevant resources.
	Web Access Center	Provides an overview of external HTTP requests and access to the web services of a host. The metrics displayed on the dashboard include the request success rate, access trends, success efficiency, distribution of accessed domain names, and other related distributions.
	Login Center	Provides an overview of the logon information of hosts. The metrics displayed on the dashboard include the geographic distribution of source and destination IP addresses, logon trends and ports, and the distribution of logon methods.

Log type	Dashboard	Description
Host log	Process Center	Provides an overview of the startup of host processes. The metrics displayed on the dashboard include the trend of process startup and the distributions of processes, process types, and specific bash or Java processes.
	Connection Center	Provides an overview of the network connections of hosts. The metrics displayed on the dashboard include the trends and distributions of network connections, source hosts, and destination hosts.
Security log	Baseline Center	Provides an overview of baseline checks. The metrics displayed on the dashboard include the distribution of pending issues, trend of new issues or resolved issues, and status of issues.
	Vulnerability Center	Provides an overview of vulnerabilities. The metrics displayed on the dashboard include the distribution of vulnerabilities, number of new vulnerabilities, number of vulnerabilities that are under verification, and number of vulnerabilities that are being fixed.
	Alert Center	Provides an overview of security alerts. The metrics displayed on the dashboard include the trend, distribution, and status of new and cleared alerts.

Billing

- You are not billed for the statistics of read and write traffic, indexing traffic, storage space, number of shards, and number of read and write operations in the dedicated Logstore. You are billed when you transform logs, ship logs, and read data from the Internet. For more information, see Log Service pricing.
- You are billed for the log analysis feature after you enable the feature in the Security Center console. For more information, see Billing.

Limits

- You can write only log data of Security Center to a dedicated Logstore.
- As required by the Cyber Security Law of the People's Republic of China, logs are retained for at least six months. We recommend that you allocate a storage capacity of 30 GB to each server.

4.9.2. Enable the log analysis feature

Before you analyze Security Center logs in the Log Service console, you must enable the log analysis feature in the Security Center console. This topic describes how to enable the log analysis feature in the Security Center console.

Prerequisites

Log Service and Security Center are activated.

Procedure

1.
2. In the left-side navigation pane, choose **Investigation > Log Analysis**.
3. In the **Activate Log Service** wizard, click **Activate Now**.
4. On the Purchase page, select the edition and specify the storage capacity. For more information about other parameters, see Purchase Security Center.

   After you specify the **log storage capacity**, the log analysis feature is enabled.

   > **Note**
   > - The log analysis feature is unavailable for Security Center of the basic edition.
   > - You can view the network logs, security logs, and host logs of Security Center of the enterprise edition.

5. Complete the payment as prompted.

   After you enable the log analysis feature, Log Service immediately collects logs from Security Center. At the same time, a dedicated project and a Logstore are automatically created and indexes are created in the Logstore.

What's next

After logs are collected, you can query, download, ship, and transform the logs. You can also create alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.9.3. Log fields

This topic describes the fields of the 14 subtypes of Security Center logs.

Network logs

- DNS logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-log-dns.
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. Multiple values are separated by vertical bars (|).
authority_num	The number of fields in the authority section.

Log field	Description
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, for example, 2018-09-25 09:59:16.
src_ip	The IP address of a source server.
src_port	The source port.

- Local DNS logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: local-dns.
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 (|).

Log field	Description
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 IP address of a host.
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
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-log-session.
asset_type	The type of an associated Alibaba Cloud service, for example, ECS.
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.

Log field	Description
session_time	The session time, for example, 2018-09-25 09:59:49.
src_ip	The IP address of a source server.
src_port	The source port.

- Web access logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-log-http.
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, for example, GET.
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.
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
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-vul-log.
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.
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
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-hc-log.
level	The level of a baseline. Valid values: low, medium, and high.
op	The action that is performed on a baseline. Valid values: ○ new: detects a new baseline. ○ verify: verifies a baseline. ○ fix: fixes 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.

Log field	Description
sub_type_name	The subtype of a baseline.
type_name	The type of a baseline.
type_alias	The type alias of a baseline.
uuid	The UUID of a client.

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
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

type_name	sub_type_name
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.
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.

Status code	Description
20	Expired.

- Security alert logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: sas-security-log.
data_source	The data source. Valid values: ○ aegis_suspicious_event: server exceptions ○ aegis_suspicious_file_v2: Webshell ○ aegis_login_log: suspicious logons ○ security_event: Security Center exceptions
level	The severity level of an alert, for example, suspicious, serious, or remind.
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.

Host logs

- Process startup logs

Log field	Description
__time__	The log time.
__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.

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

- Process snapshot logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-snapshot-process.
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.

Log field	Description
start_time	The time when a process starts.
user	The username.
uid	The ID of a user.

- Logon logs

> **Note** The logon attempts within 1 minute are recorded in one log entry. The warn_count field indicates the number of logon attempts.

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-log-login.
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: Secure Shell (SSH) logon ○ RDPLOGIN: remote desktop logon ○ IPCLOGIN: IPC logon
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
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-log-crack.
uuid	The UUID of a client.
ip	The IP address of a client.
warn_ip	The IP address of a source server.

Log field	Description
warn_port	The logon port.
warn_type	The type of a logon. Valid values: ○ SSHLOGIN: SSH logon ○ RDPLOGIN: remote desktop logon ○ IPCLOGIN: IPC logon
warn_user	The logon username.
warn_count	The number of failed logon attempts.

- Network connection logs

  Note The changes in network connections are collected on the host every 10 seconds to 1 minute. The logs of network connections in some states are collected.

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-log-network.
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.
proc_path	The path of a process file.
proto	The protocol that is used to establish a network connection, for example, UDP or raw (raw socket).
status	The connection status. For more information, see Status codes of network connections.

  Status codes of network connections

Status code	Description
1	closed

Status code	Description
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 snapshot logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-snapshot-port.
uuid	The UUID of a client.
ip	The IP address of a client.
proto	The protocol that is used to establish a network connection, for example, TCP, UDP or raw (raw socket).
src_ip	The IP address that is listened on.
src_port	The port that is listened on.
pid	The ID of a process.
proc_name	The name of a process.

- Account snapshot logs

Log field	Description
__time__	The log time.
__topic__	The topic of a log entry. Valid value: aegis-snapshot-host.

Log field	Description
uuid	The UUID of a client.
ip	The IP address of a client.
user	The username of an account.
perm	Indicates whether a user has root permissions. ○ 0: The user does not have root permissions. ○ 1: The user has root permissions.
home_dir	The home directory of a user.
groups	The group to which a user belongs.
last_chg	The date when a password is last modified.
shell	The shell commands.
domain	The Windows domain.
tty	The logon terminal.
warn_time	The notification date for password expiration.
account_expire	The date when an account expires.
passwd_expire	The date when a password expires.
login_ip	The IP address of the last remote logon client.
last_logon	The date and time of the last logon.
status	The status of a user. ○ 0: disabled ○ 1: normal

4.10. WAF logs

4.10.1. Usage notes

Web Application Firewall (WAF) allows you to query, analyze, transform, and consume logs by using Log Service. After you enable the log analysis feature, the access logs and anti-attack logs of your website domain are collected in real time. This feature helps you better protect and manage your website. This topic describes the assets, billing, and limits of using the log analysis feature.

Assets

- Dedicated projects and Logstores

- If you set Region to Mainland China when you purchased the WAF instance, after you enable the log analysis feature, Log Service creates a project named waf-project-Alibaba Cloud account ID-cn-hangzhou and a Logstore named waf-logstore.
- If you set Region to International when you purchased the WAF instance, after you enable the log analysis feature, Log Service creates a project named waf-project-Alibaba Cloud account ID-ap-southeast-1 and a Logstore named waf-logstore.

- Dedicated dashboards

  After you enable the log analysis feature, Log Service creates three dashboards by default.

  > **Note** To ensure that the dashboards are up to date, we recommend that you do not modify the dashboards. You can create a custom dashboard to visualize the results of log analysis. For more information, see Create a dashboard.

Dashboard	Description
Operation Center	The Operation Center dashboard shows the details of website operation, traffic, and attacks. The metrics of website operation include Valid Request Ratio, Valid Request Traffic Ratio, Peak Attack Size, Attack Traffic, and Attack Count. The traffic metrics include Peak Network In, Peak Network Out, Received Requests, Traffic Received, and Traffic Out.
Access Center	The Access Center dashboard shows the basic access details, the access trend, the distribution of visitors, and other information. The metrics of basic access details include the number of page views (PVs) and the number of unique visitors (UVs).
Security Center	The Security Center dashboard shows the attack metrics, attack types, attack trend, attacker distribution, and other information.

Billing

You are charged based on the log storage capacity and duration that you select on the buy page of WAF. You are charged by Log Service when you transform logs, ship logs, and read data on the Internet. For more information, see Log Service Pricing.

Limits

- To use the log analysis feature, you must ensure that the billing method is subscription and the edition of WAF is Pro, Business, or Enterprise.
- If you have overdue payments in Log Service, you can no longer use the log analysis feature.
- Only the data that is generated in WAF can be written to the dedicated Logstore. This limit does not apply to other log operations such as query, statistics, alerts, and consumption.
- The dedicated Logstore cannot be deleted. The data retention period of the dedicated Logstore cannot be changed.
- You must ensure that the available storage space of WAF logs is sufficient. After the log storage capacity is exhausted, logs can no longer be stored.

> **Note** You can view the usage of log storage space in the WAF console. However, the usage is not updated in real time. The displayed usage does not include the usage in the last two hours.

Benefits

- Classified protection compliance: The website access logs are stored for more than six months. Requirements of classified protection compliance are met.
- Ease of use: To enable the log analysis feature, you only need to perform a few simple operations. The feature ensures that the access logs and anti-attack logs of your website domain are collected in real time. You can customize the log storage capacity and duration. You can select a website for log collection.
- Real-time analysis: The WAF console provides the real-time log analysis service and out-of-the-box dashboards. The dashboards provide insights into the visits to and attacks on your website.
- Real-time monitoring: You can monitor your website by using specific metrics. You can customize alert settings to receive alerts almost in real time. This allows you to handle the exceptions of critical business in a timely manner.
- Integration: You can use Log Service together with other data solutions such as real-time compute, cloud storage, and visualization to maximize the value of your business data.

Scenarios

- Track anti-attack logs and trace the source of security threats.
- Monitor web requests in real time and view traffic trends.
- Obtain information about the efficiency of security operations and respond to issues in a timely manner.
- Export security logs to data centers.

4.10.2. Enable the log analysis feature

This topic describes how to enable the log analysis feature in the Web Application Firewall (WAF) console. After you enable the log analysis feature, you can collect the logs of WAF to Log Service.

Prerequisites

- A subscription-based WAF instance is created. The edition of WAF is Pro, Business, or Enterprise. For more information, see Activate Alibaba Cloud WAF.

  > **Note**
  > - On the buy page of WAF, set **Access Log Service** to **YES**, and specify the period for which you want to store WAF logs, and specify the maximum size of stored logs.
  > - If you have activated the Basic Edition, you must upgrade WAF to the Pro Edition, Business Edition, or Enterprise Edition before you enable the log analysis feature. For more information, see Renewal and upgrade of subscription WAF.

- Your website is added to WAF. For more information, see Add your website to WAF.

Procedure

1.
2.
3. Authorize WAF to use the AliyunWAFAccessingLogRole role to access Log Service as prompted.

   > Note
   > - This operation is required only when you enable the log analysis feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
   > - If you use a RAM user to log on to WAF, you must grant required permissions to the RAM user. For more information, see RAM user authorization.
   > - To ensure that WAF logs can be collected to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

4. On the **Log Service** page, click **Open now**.
5. On the **Log Service** page, select the domain name of your website that is protected by WAF, and turn on the **Status** switch to enable the log analysis feature.

What's next

After the logs of WAF are collected to Log Service, you can query, analyze, download, ship, and transform the collected logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.10.3. Manage the log storage space

This topic describes how to manage the log storage space in the Web Application Firewall (WAF) console.

Context

1. Log on to the Web Application Firewall console .
2. In the left-side navigation pane, choose **Log Management**>**Log Service** .
3. On the **Log Service** page, perform the following operations to manage the log storage space:
   - View the log storage space.

     Note You can view the usage of log storage space in the WAF console. However, the usage is not updated in real time. The displayed usage does not include the usage in the last two hours. If the usage of log storage space reaches 70%, upgrade the storage capacity to make sure that new logs can be stored.
   - Upgrade the log storage capacity: Click **Upgrade Storage** and select a larger storage capacity.
   - Clear the log storage space: Click **Clear** to clear all the log entries in your log storage space.

     After you enable the log analysis feature, you can clear your log storage space for four times in total.

     Warning Log entries that are cleared cannot be restored. Clear log entries with caution.

4.10.4. Log fields

This topic describes the fields of website access, attack, and protection logs.

Log field	Description
__topic__	The topic of the log. The value is fixed as waf_access_log.
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.
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.

Log field	Description
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.
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.

Log field	Description
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.
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.

Log field	Description
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.
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.

Log field	Description
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).
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.

Log field	Description
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.
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.

Log field	Description
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.
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.

Log field	Description
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.
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.
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.
status	The HTTP status code that is returned by WAF to the client.
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.

Log field	Description
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.
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.

Log field	Description
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.

4.11. Cloud Firewall logs

4.11.1. Usage notes

The log analysis feature of Cloud Firewall allows you to collect, query, analyze, transform, and consume logs of Internet traffic in real time. The log analysis feature also ensures that your resources meet the requirements of classified protection compliance. This topic describes the assets, billing, and limits of the log analysis feature in Cloud Firewall.

Assets

- Dedicated projects and dedicated Logstores

  After you enable the log analysis feature of Cloud Firewall, Log Service creates a project named cloudfirewall-project-Alibaba Cloud account ID-ap-southeast-1 and a Logstore named cloudfirewall-logstore.

  > Note

- Dedicated dashboards

  After you enable the log analysis feature, Log Service generates one dashboard by default.

  > Note We recommend that you do not make changes to the dedicated dashboard because this may affect the usability of the dashboard. You can create a custom dashboard to visualize the results of log analysis. For more information, see Create a dashboard.

Dashboard	Description
Report	Shows the statistics of Cloud Firewall. The information displayed on the dashboard includes the basic indicators, inbound traffic sources, outbound traffic distribution, and system stability.

Billing

You are billed based on the duration and storage space when you enable the log analysis feature in the Cloud Firewall console. You are also billed when you transform logs, ship logs, and read data on the Internet in the Log Service console. For more information, see Log Service pricing.

Limits

- You can write only Cloud Firewall log data to a dedicated Logstore.
- You cannot delete dedicated Logstores. You cannot modify the data retention period of a dedicated Logstore.
- If you have overdue payments for your Log Service resources, the log analysis feature is automatically stopped. To ensure service continuity, you must pay your overdue payment within the prescribed time limit.
- If the storage space of a dedicated Logstore is full, no more logs can be written to the Logstore.

> **Note** You can view the usage of log storage space in the Cloud Firewall console. However, the usage is not updated in real time. The storage space usage displayed is delayed by two hours.

Benefits

- Classified protection compliance: The log analysis feature can store the website access logs for more than six months. The log analysis feature ensures that websites meet the requirements of classified protection compliance.
- Simple configuration: You can perform a few simple steps to enable the analysis feature and start to collect Internet traffic logs in real time.
- Real-time analysis: Based on Log Service, the log analysis feature provides real-time analysis and log reports for Internet traffic and access details.
- Real-time alerts: The log analysis feature can be used to monitor specified indicators and send alerts in real time. This helps you resolve business exceptions at the earliest opportunity.
- High compatibility: The log analysis feature is compatible with other data solutions such as stream processing, cloud storage, and visualization. This allows you to maximize the value of your business data.

4.11.2. Enable the log analysis feature

This topic describes how to enable the log analysis feature in the Cloud Firewall console. After you enable the log analysis feature, you can collect Internet traffic logs to Log Service.

Procedure

1. Log on to the Cloud Firewall console.
2. In the left-side navigation pane, choose **Log Analysis > Log Analysis**.
3. Click **Activate Now**.
4. Purchase the Cloud Firewall service and complete the payment as prompted.

   In the **Log Analysis** section, click **Yes**. Then, specify a log storage size based on your business requirements. For information about other parameters, see Purchase Cloud Firewall.
5. On the **Log Analysis** page, select **internet_log** and turn on the status switch.

> **Notice** If the storage space usage exceeds 70%, we recommend that you upgrade the log storage space. Otherwise, you cannot store additional logs and you may lose data.

What's next

After Internet traffic logs are collected to Log Service, you can query, analyze, download, ship, and transform the collected logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.11.3. Manage log storage space

This topic describes how to manage the log storage space in the Cloud Firewall console. You can view, upgrade, and clear the log storage space that you purchase.

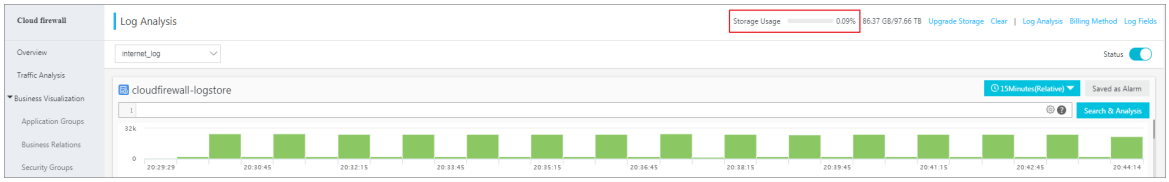
Procedure

1. Log on to the Cloud Firewall console.
2. In the left-side navigation pane, choose **Log Analysis > Log Analysis**.
3. On the **Log Analysis** page, perform the following operations as needed:
   - View the log storage space.

     > **Note** You can view the usage of log storage space in the Anti-DDoS Pro or Anti-DDoS Premium console. However, the usage is not updated in real time. The storage space usage displayed is delayed by two hours. If the storage space usage exceeds 70%, you must upgrade the log storage space. Otherwise, you cannot store additional logs.

   - Upgrade the storage space: Click **Upgrade Storage** and select a log storage capacity.
   - Clear log data: Click **Clear** to clear log data.

     > **Warning** You cannot restore log data that is cleared. Proceed with caution.



4.11.4. Log fields

This topic describes the fields of the log entries for Internet traffic.

Log field	Description
__topic__	The topic of a log entry. Valid value: cloudfirewall_access_log.
log_type	The type of a log entry. Valid value: internet_log. This value indicates a log entry for Internet traffic.
aliuid	The ID of an Alibaba Cloud account.

Log field	Description
app_name	The name of the protocol over which an application is accessed. The value can be HTTPS, NTP, SIP, SMB, NFS, or DNS. If the protocol is unknown, the value Unknown is displayed.
direction	The direction of Internet 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 when 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: packet/s.
ip_protocol	The type of an IP protocol. Valid values: TCP and UDP.
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: packet/s.
region_id	The region from which access traffic is originated, for example, cn-beijing.
rule_result	The result of how an access policy processes Internet traffic. Valid values: • pass • alert • drop
src_ip	The IP address of a source server.
src_port	The source port of a host that sends traffic data.
start_time	The time when a session starts. The value is a UNIX timestamp. Unit: seconds.

Log field	Description
start_time_min	The time when 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 packets. Unit: bit/s.
total_packet_bytes	The total size of inbound and outbound packets. Unit: bytes.
total_packet_count	The total number of packets.
total_pps	The total rate of inbound and outbound packets. Unit: packet/s.
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 of a resource 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 that is matched.
ips_rule_name_en	The name of an IPS that is matched.
attack_type_name	The Chinese name of an attack type.
attack_type_name_en	The name of an attack type.

4.12. Layer 7 access logs for SLB

4.12.1. Usage notes

Alibaba Cloud Server Load Balancer (SLB) provides the access log management feature that is supported by Log Service. You can use SLB access logs to analyze user behavior and the distribution of users and troubleshoot issues. This topic describes the resources, billings, and limits that are related to the access log management feature.

Resources

- Project and Logstore

- The indexing feature is automatically enabled for the Logstore. Indexes are configured for some fields.
- By default, log data is permanently retained in the Logstore. You can modify the retention period of log data based on your business requirements. For more information, see Manage a Logstore.

> **Note** You must not delete the project or Logstore that is related to the log management feature. Otherwise, the access logs cannot be sent to Log Service.

- Dedicated dashboards

  After you enable the feature, two dedicated dashboards are automatically created for SLB access logs.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize the results of log queries. For more information, see Create a dashboard.

Dashboard	Description
slb_layer7_operation_center_en	Displays the overall operational statistics of SLB, including page views (PVs), unique views (UVs), the request success rate, size of request messages, and size of response messages.
slb_layer7_access_center_en	Displays the details of requests sent to SLB, including the distribution of clients, trend of request methods, trend of status codes, top clients, and topology of request messages.

Limits

- The access log management feature is available only in SLB instances for which a layer-7 listener is configured.
- The project that stores SLB access logs must reside in the same region as the SLB instance.

Billing

- You are not charged for using the access log management feature of SLB.
- After SLB access logs are shipped to Log Service, you are charged for the storage space that the logs occupy and the number of read/write operations on the logs. You are also charged for reading, transforming, and shipping the logs. For more information, see Billing methods.

Benefits

- Easy to use: The access log management feature allows developers and operations and maintenance (O&M) personnel to unburden from tedious and time-consuming log processing. They can focus on business development and technical research.
- Capable of processing large amounts of data: The size of access logs is proportional to the request PVs of SLB instances. Processing large amounts of data requires high performance and may introduce high costs. Log Service allows you to analyze 100 million log entries in 1 second and is cost-competitive compared with open source solutions.
- Capable of processing data in real time: Timeliness is essential in multiple scenarios such as DevOps, data monitoring, and alerting. The access log management feature of SLB integrates the data

processing capabilities of Log Service. Large amounts of log data can be processed within seconds.

- Flexible: You can enable or disable the access log management feature based on the specification of your SLB instances. You can also set a custom retention period for log data. In addition, the storage capacity of a Logstore automatically scales based on your the data volume.

4.12.2. Enable the access log management feature

This topic describes how to enable the access log management feature in the SLB console. After you enable the feature, you can use Log Service to collect SLB access logs.

Prerequisites

- An SLB instance is created. For more information, see Create a CLB instance.
- An HTTP or HTTPS listener is configured for the SLB instance. For more information, see Add an HTTP listener or Add an HTTPS listener.
- A project and a Logstore are created in the region where the SLB instance resides. For more information, see Create a project and a Logstore.

Procedure

1. Log on to the SLB console.
2. In the upper-left corner of the page, select the region where the SLB instance resides.
3. In the left-side navigation pane, choose **Logs > Access Logs**.
4. Authorize SLB to assume the AliyunLogArchiveRole role to access Log Service.

   > **Note**
   > - If you have authorized SLB to assume the AliyunActionTrailDefaultRole role, skip this step.
   > - You must not delete the RAM role or revoke the permissions from the RAM role. Otherwise, logs cannot be shipped to Log Service.
   > - If you use a RAM user to log on to SLB, you must authorize the RAM user by using an Alibaba Cloud account. For more information, see Authorize a RAM user to use the access log feature.

5. On the **Access Logs (Layer-7)** page, click **Configure** in the Actions column of the instance.
6. In the **Configure Logging** dialog box, select an available project and a Logstore. , and then click **OK**.

   After you complete the configuration, indexes are automatically created for the data in the selected Logstore. If indexes were created in the Logstore, the indexes are overwritten.

What's next

After SLB access logs are collected by Log Service, you can query, download, ship, and transform the logs. You can also create alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.12.3. Log fields

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

Field	Description
__topic__	The topic of the log entry. Valid value: slb_layer7_access_log.
body_bytes_sent	The size of the HTTP response message body sent to the client. Unit: bytes.
client_ip	The IP address of the client.
host	The IP address of the 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 header in the request message.
http_referer	The HTTP referer header in the request message received by the proxy.
http_user_agent	The HTTP user-agent header in the request message received by the proxy.
http_x_forwarded_for	The x-forwarded-for content in the request message received by the proxy.
http_x_real_ip	The real IP address of the client.
read_request_time	The time when the proxy reads the request message. Unit: milliseconds.
request_length	The length of the request message, which includes the startline, 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 request URI received by the proxy.
scheme	The request schema. Valid values: http and https.
server_protocol	The HTTP version received by the proxy, for example, HTTP/1.0 or HTTP/1.1.
slb_vport	The listening port of the SLB instance.
slbid	The ID of the SLB instance.

Field	Description
ssl_cipher	The cipher suite used to establish an SSL connection, for example, ECDHE-RSA-AES128-GCM-SHA256.
ssl_protocol	The protocol used to establish an SSL connection, such as TLSv1.2.
status	The HTTP status code sent from the proxy.
tcpinfo_rtt	The round-trip time (RTT) of TCP packets. Unit: milliseconds.
time	The time when the log entry is generated.
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 received by the proxy from the backend server.
vip_addr	The virtual IP address.
write_response_time	The response duration of the proxy. Unit: milliseconds.

4.13. Layer 4 monitoring metrics for SLB

4.13.1. Usage notes

The Layer 4 monitoring feature is available for Server Load Balancer (SLB) instances and the monitoring metrics can be sent to Log Service. You can view the traffic, queries per second (QPS), and error rate of SLB instances by using monitoring metrics that are accurate to the second. This improves fine-grained service monitoring and troubleshooting.

Assets

- Custom projects and Metricstores

  > **Note** We recommend that you do not delete the projects or Metricstores that are related to Layer 4 monitoring metrics for SLB instances. Otherwise, the monitoring metrics cannot be sent to Log Service.

- Dedicated dashboards

  By default, Log Service generates two dedicated dashboards for monitoring metrics that are accurate to the second.

> **Note** We recommend that you do not make changes to the dedicated dashboards. This may affect the usability of the dashboards. You can create a custom dashboard to visualize data query and analysis results. For more information, see Create a dashboard.

Dashboard	Description
SLB Layer 4 Monitoring	Shows the trends of metrics such as SLB instances, ports, outbound traffic, inbound traffic, and the number of connections.
SLB Layer 4 Data Analysis	Shows the statistics of metrics such as SLB instances, ports, outbound traffic, inbound traffic, and the number of connections.

Billing

- You are not charged for the Layer 4 monitoring feature in the SLB console.
- After the monitoring metrics for SLB instances are sent to Log Service, you are charged based on storage space, read traffic, and the number of requests. You are also charged when you transform and ship data. For more information, see Log Service pricing.

Limits

- Only an SLB instance that is configured with a TCP or UDP listener supports the Layer 4 monitoring feature.
- The project that is used to store monitoring metrics must reside in the same region as the specified SLB instance.

4.13.2. Enable the Layer 4 monitoring feature

This topic describes how to enable the Layer 4 monitoring feature for a Server Load Balancer (SLB) instance in the SLB console. After you enable the feature, Layer 4 monitoring metrics are sent to Log Service. The monitoring metrics are accurate to the second.

Prerequisites

- An SLB instance is created. For more information, see Create a CLB instance.
- A TCP or UDP listener is configured for the SLB instance. For more information, see Add a TCP listener and Add a UDP listener.
- A Log Service project and a Metricstore are created in the region where the SLB instance resides. For more information, see Create a project and Create a Metricstore.

Procedure

1. Log on to the SLB console.
2. In the left-side navigation pane, choose **CLB (FKA SLB) > Instances**.
3. On the Instances page, click the ID of the SLB instance.
4. Click **Fine-grained Monitoring**.
5. Enable the fine-grained monitoring feature in the current region.

   If the fine-grained monitoring feature is enabled in the current region, skip this step.

   i. Click **Enable Fine-grained Monitoring**.

ii. In the **Enable Fine-grained Monitoring for SLB** dialog box, select the project and Metricstore that you created, select the **I understand the above information** check box, and then click **OK**.

> **Notice**
>
> - When you perform this operation, the system automatically creates a Resource Access Management (RAM) role named AliyunServiceRoleForSlbLogDelivery. You can asign this RAM role to SLB to send monitoring metrics that are accurate to the second to Log Service.
> - To ensure that monitoring metrics for SLB instances can be sent to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

6. Enable the Layer 4 monitoring feature for the TCP or UDP listener.
   i. On the Fine-grained Monitoring tab, click **Settings**.
   ii. On the **Settings** tab, turn on the **Fine-grained Monitoring** switch for the TCP or UDP listener.

What's next

After Log Service collects Layer 4 monitoring metrics for the specified SLB instance, you can query, analyze, download, ship, and transform the metrics in the Log Service console. You can also configure alerts for the metrics. For more information, see Common operations on logs of Alibaba Cloud services.

4.13.3. Layer 4 monitoring metrics

This topic describes the Layer 4 monitoring metrics for Server Load Balancer (SLB) instances. The monitoring metrics are accurate to the second.

The metrics in this topic use the format of time series data. For more information, see Metric. You can query and analyze metrics by using PromQL or SQL query language. For more information, see Overview of time series search and analysis.

Metrics

Metric	Description
actConnsPS	The number of active connections per second.
connsPS	The number of new connections per second.
dropConnPS	The number of dropped connections per second.
failConnPS	The number of failed connections per second.
inActConnPS	The number of inactive connections per second.
inBitsPS	The number of inbound bits per second. Unit: bit/s.
inDropBitsPS	The number of inbound bits that are dropped per second. Unit: bit/s.

Metric	Description
inDropPktsPS	The number of inbound packets that are dropped per second.
inPktsPS	The number of inbound packets per second.
aclDropBitsPS	The number of inbound bits that are dropped by access control lists (ACLs) per second. Unit: bit/s.
aclDropPktsPS	The number of inbound packets that are dropped by ACLs per second.
maxConnsPs	The number of concurrent connections per second.
outBitsPS	The number of outbound bits per second. Unit: bit/s
outDropBitsPS	The number of outbound bits that are dropped per second. Unit: bit/s.
outDropPktsPS	The number of outbound packets that are dropped per second.

Labels

Label	Description
lbId	The ID of an SLB instance.
listenerId	The ID of a listener.
protocol	The protocol of a listener.
vip	The IP address that is listened on.
vport	The port that is listened on.

4.14. VPC flow logs

4.14.1. Usage notes

Virtual Private Cloud (VPC) provides the flow log feature that is supported by Log Service. You can use this feature to record the traffic that is transferred over the elastic network interfaces (ENIs) and vSwitches in a VPC. You can also use this feature to record the inbound and outbound traffic of a VPC. The feature allows you to check access control rules, monitor network traffic, and fix network errors. This topic describes the resources, billing methods, and limits that are related to the flow log feature of VPC.

The traffic information that is captured by the flow log feature is written as log data to Log Service. Each log entry includes a five-tuple traffic flow captured within a specified time window. The maximum time window is about 10 minutes. During this time window, the flow log feature captures and aggregates traffic flows and then sends the traffic flows as log entries to Log Service. If you create a flow log instance for a VPC or a vSwitch, the traffic that is transferred over the ENIs in the VPC or vSwitch is captured. These ENIs include the ENIs that are attached to the vSwitch after the flow log instance is created.

Assets

- Projects and Logstores

  > **Note**
  > - We recommend that you do not delete the projects or Logstores that are related to VPC flow logs. Otherwise, the flow logs cannot be sent to Log Service.
  > - After the VPC flow log feature is enabled, the data retention period of the Logstores that store VPC flow logs must be changed to seven days.

- Dedicated dashboards

  Log Service generates three dashboards for VPC flow logs by default.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
Logstore Name-vpc_flow_log_traffic_cn	Displays the overall traffic flows of a VPC. The information includes the heatmap that shows the number of bytes that are sent from the source. The information also includes the top 10 destination ports by number of accepted bytes, and the number of bytes that are sent per minute by various protocols.
Logstore Name-vpc_flow_log_rejection_cn	Displays the traffic flows that are rejected by security groups and network ACLs. The information includes the total number of bytes in rejected packets, percentage of rejected bytes, and percentage of rejected packets.
Logstore Name-vpc_flow_log_overview_cn	Displays the overall information of a VPC. The information includes the total number of actions about traffic flows, accepted bytes, rejected bytes, and accepted packets.

Billing

The flow log feature allows you to ship only the captured flow logs to Log Service. The fees related to this feature include the log capture fee and the usage fee of Log Service resources.

- Log capture fee

  The log capture fee is calculated based on the number of the captured logs.

> **Note**
> - You are not charged for capturing flow logs during the public preview period.
> - Bills are generated in the VPC console.

- The usage fee of Log Service resources

  After VPC flow logs are collected by Log Service, you are charged based on the storage space, read traffic, the number of requests, data transformation, and data shipping. For more information, see Log Service Pricing.

Limits

- Supported regions
  - The project that stores VPC flow logs must reside in the same region as the flow log instance.
  - The flow log feature is available for public preview. To use this feature, you can submit a ticket.

    The following table lists the regions where the flow log feature is supported.

Parameter	Supported regions
Asia Pacific	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), Japan (Tokyo), Singapore (Singapore), Australia (Sydney), Malaysia (Kuala Lumpur), and Indonesia (Jakarta)
Europe & America	US (Silicon Valley), US (Virginia), Germany (Frankfurt), and UK (London)
Middle East & India	India (Mumbai) and UAE (Dubai)

- Resource limits

Item	Limit	Adjustable
Maximum number of flow log instances that can be created in a region	10	Submit a ticket.

Item	Limit	Adjustable
VPCs that do not support flow logs	VPCs that contain instances of the following instance families:	Note If the VPC to which a specified vSwitch or ENI belongs contains ECS instances, you must upgrade the ECS instances in two conditions. One condition is that the ECS instances are of the specified instance families. The other condition is that you have created flow log instances. If you do not upgrade the ECS instances in these two conditions, the flow log feature may not function as normal.
vSwitches that do not support the flow log feature	VPCs to which the vSwitches belong contain ECS instances of the following instance families:	
ENIs that do not support flow logs	VPCs to which the ENIs belong contain ECS instances of the following instance families:	

4.14.2. Enable the flow log feature

This topic describes how to enable the flow log feature in the Virtual Private Cloud (VPC) console. After you enable the feature, you can use Log Service to collect flow logs.

Prerequisites

- An elastic network interface (ENI), a VPC, or a vSwitch is created. For more information, see Create an ENI, Create and manage a VPC, and Work with vSwitches.
- A project and a Logstore are created in the region where the resource instances reside. For more information, see Create a project and Create a Logstore.

Procedure

Note Before you can use a RAM user to enable the flow log feature, you must grant the required permissions to the RAM user. For more information, see RAM user authorization.

1.
2.
3. The first time you enable the flow log feature, you must click **Authorize** and complete the authorization as prompted.

VPC flow logs can be written to Log Service only after you complete the authorization.

> **Notice** You cannot delete the RAM role or revoke the required permissions from the RAM role. Otherwise, flow logs cannot be delivered to Log Service.

4. In the top navigation bar, select the region where the instances reside.

   For more information about the regions that support the flow log feature, see Feature release and supported regions.

5. On the **Flow Log** page, click **Create FlowLog**.
6. In the **Create FlowLog** dialog box, set the parameters and click **OK**. The following table describes the parameters.

Parameter	Description
Flow Log Name	The name of the flow log instance. The name must be 2 to 128 characters in length, and can contain digits, hyphens (-), and underscores (_). The name must start with a letter.
Resource Type	Select the type of the resource from which you want to capture traffic, and then select a resource. Valid values: ○ **VPC**: captures traffic from all ENIs in a specified VPC. If the VPC contains Elastic Compute Service (ECS) instances that do not support flow logs, traffic information about the ENIs of the ECS instances cannot be captured. ○ **VSwitch**: captures traffic from all ENIs associated with a specified vSwitch. If the vSwitch contains ECS instances that do not support flow logs, traffic information about the ENIs of the ECS instances cannot be captured. ○ **Network Interface**: captures traffic information about a specified ENI. If the ENI is associated with an ECS instance that does not support flow logs, traffic information about the ENI cannot be captured. ECS instances of the following types do not support flow logs: To use the flow log feature, you must upgrade your ECS instances. For more information, see Upgrade the instance types of subscription instances and Change the instance type of a pay-as-you-go instance.
Resource Instance	Select a resource instance from which you want to capture traffic.
Traffic Type	The type of traffic. ○ **All**: captures traffic of the specified resource. ○ **Allow**: captures traffic that is allowed by the security group rules of the specified resource. ○ **Drop**: captures traffic that is denied by the security group rules of the specified resource.

Parameter	Description
Project	Select a Log Service project that is used to manage resources related to VPC flow logs, such as Logstores and dashboards. ○ **Select Project**: Select an existing project. ○ **Create Project**: Create a project. For more information, see Create a project.
Logstore	Select a Logstore that is used to store VPC flow logs. ○ **Select Logstore**: Select an existing Logstore. ○ **Create Logstore**: Create a Logstore. For more information, see Create a Logstore.
Turn on FlowLog Analysis Report Function	If you turn on this switch, Log Service enables the indexing feature for the Logstore and creates a dashboard. After indexing is enabled, you can query and analyze VPC flow logs.
Description	The description of the flow log instance. The description must be 2 to 256 characters in length and cannot start with `http://` or `https://`.

Result

What's next

After Log Service collects VPC flow logs, you can query, analyze, download, ship, and transform the logs. You can also create alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.14.3. Log fields

This topic describes the fields of VPC flow logs.

Field	Description
__topic__	The topic of the log entry. Valid value: flow_log.
version	The version of the flow log.
vswitch-id	The ID of the vSwitch to which the ENI is attached.
vm-id	The ID of the ECS instance to which the ENI is attached.
vpc-id	The ID of the VPC to which the ENI belongs.
account-id	The ID of the Alibaba Cloud account.
eni-id	The ID of the ENI.

Field	Description
srcaddr	The source IP address.
srcport	The source port.
dstaddr	The destination IP address.
dstport	The destination port.
protocol	The IANA protocol number of the traffic. For more information, see Internet protocol number.
direction	The direction of the traffic flow. Valid values: • in: inbound traffic • out: outbound traffic
packets	The number of data packets.
bytes	The number of bytes in a data packet.
start	The start time of the capture window.
end	The end time of the capture window.
log-status	The logging status of the flow log. Valid values: • OK: Flow log data is recorded as expected. • NODATA: No inbound or outbound traffic is transmitted over the ENI during the capture window. • SKIPDATA: Some flow log data is not recorded within the capture window.
action	The actions performed on traffic flows. Valid values: • ACCEPT: the traffic that security groups allow to transfer. • REJECT: the traffic that security groups disallow to transfer.

4.15. EIP logs

4.15.1. Overview

Alibaba Cloud allows you to integrate Elastic IP Address (EIP) with Log Service to implement fine-grained per-second monitoring. Logs that contain fine-grained monitoring data of network bandwidth are delivered to Log Service. This feature allows you to monitor the fluctuations of Internet data transfer in real time. You can adjust the peak bandwidth of an EIP in a timely manner. This topic describes EIP log-related resources, billing, and limits.

Resources

- Projects and Logstores

> **Note**
> - You must not delete the projects or Logstores that are related to EIP logs. Otherwise, EIP logs cannot be sent to Log Service.
> - After the EIP per-second monitoring feature is enabled, the data retention period of Logstores that store EIP logs is forcibly changed to 7 days.

- Dedicated dashboard

  By default, only one dedicated dashboard is created.

  > **Note** We recommend that you do not make changes to the dedicated dashboard because this may affect the availability of the dashboard. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
eip_monitoring	Allows you to monitor the fluctuations of Internet data transfer in real time. These fluctuations include the peak inbound and outbound bandwidth per second, inbound and outbound packet rates per second, inbound and outbound packet loss rates per seconds, and inbound and outbound TCP session establishment rates.

Billing

- You are not charged when you use the log feature of EIP.
- After EIP logs are shipped to Log Service, you are charged for the storage space that the log data occupies and the number of read/write operations. You are also charged when Log Service reads, transforms, and ships the data. For more information, see Billing methods.

Limits

- The Log Service project that stores EIP logs must reside in the same region as the elastic IP address (EIP).
- You can use an Alibaba Cloud account to enable the fine-grained monitoring feature for a maximum of 10 EIPs.

  If you need to enable the fine-grained monitoring feature for more EIPs, you must submit a ticket.

4.15.2. Enable fine-grained monitoring

The fine-grained monitoring feature allows you to collect logs that contain monitoring data to Log Service. This topic describes how to enable the fine-grained monitoring feature for an elastic IP address (EIP) in the Virtual Private Cloud (VPC) console.

Prerequisites

- An EIP is created. For more information, see Apply for an EIP.
- A Log Service project and Logstore are created in the region where the EIP resides. For more information, see Step 2: Create a project and a Logstore.

Procedure

1. Log on to the VPC console.
2. In the top navigation bar, select the region where the EIP resides.
3. In the left-side navigation pane, choose **Access to Internet > Internet Tool Kit**.
4. On the **Internet Tool Kit** page, click **Fine-grained Monitoring**.
5. On the **High Definition Traffic Monitor** page, follow the on-screen instructions to complete authorization.

   Authorize VPC to assume the AliyunVPCLogArchiveRole role to access Log Service resources.

   > **Note**
   > - This operation is required the first time that you enable the feature. You must complete the authorization by using your Alibaba Cloud account.
   > - If you use a RAM user to enable the fine-grained monitoring feature, you must make sure that the RAM user is granted the required permissions. For more information, see RAM user authorization.
   > - Do not revoke the permissions from the RAM role or delete the RAM role. If you revoke the permissions from the RAM role or delete the RAM role, the fine-grained monitoring data of the EIP cannot be sent to Log Service.

6. On the **High Definition Traffic Monitor** page, find the EIP for which you want to enable the fine-grained monitoring feature and click **Enable Fine-grained Monitoring** in the Actions column.
7. In the **Log Settings** panel, select the Log Service project and Logstore from the drop-down lists and click **OK**.

What's next

After EIP logs are collected to Log Service, you can query, analyze, download, ship, and process these logs. You can also configure alerts for these logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.15.3. Log fields

This topic describes the fields of Elastic IP Address (EIP) logs.

Field	Description
__topic__	The topic of the log. Valid value: eip.
type	The details of peak rates per second.
tid	The ID of the sender.
time	The sending time.
gw_ip	The IP address of a gateway.
eip	The IP address of an EIP.
in_Bps	The ingress bandwidth. Unit: bytes/s

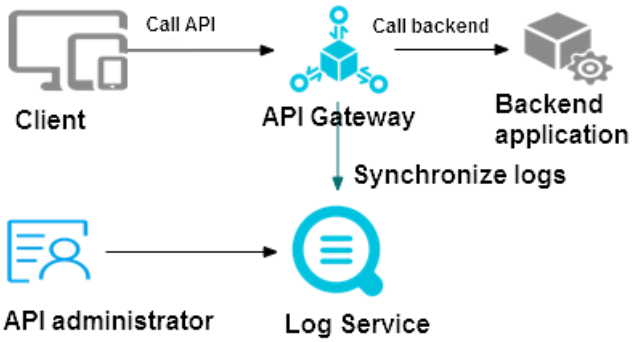
Field	Description
out_Bps	The egress bandwidth. Unit: bytes/s
in_pps	The inbound packet rate. Unit: packets per second (pps).
out_pps	The outbound packet rate. Unit: pps.
In_syn_speed	The inbound TCP session establishment rate. Unit: pps.
out_syn_speed	The outbound TCP session establishment rate. Unit: pps.
In_syn_ack_speed	The inbound TCP session acknowledgement rate. Unit: pps.
out_syn_ack_speed	The outbound TCP session acknowledgement rate. Unit: pps.
In_fin_speed	The inbound TCP session termination rate. Unit: pps.
out_fin_speed	The outbound TCP session termination rate. Unit: pps.
In_rst_speed	The inbound TCP session re-establishment rate. Unit: pps.
out_rst_speed	The outbound TCP session re-establishment rate. Unit: pps.
out_ratelimit_drop_speed	The outbound packet loss threshold. Unit: pps.
in_ratelimit_drop_speed	The inbound throttling packet loss rate. Unit: pps.
out_drop_speed	The outbound throttling packet loss rate. Unit: pps.
in_drop_speed	The inbound packet loss rate. Unit: pps.
timestamp	The timestamp. Unit: ms.

4.16. API Gateway access logs

4.16.1. Usage notes

Alibaba Cloud API Gateway provides the log management feature that is supported by Log Service. You can use this feature to ship API Gateway access logs to Log Service in real time. Log Service allows you to query, transform, and consume the data in real time. This topic describes the resources, billings, and limits that are related to the log management feature.

API Gateway provides the API hosting service to facilitate microservice aggregation, frontend and backend separation, and system integration. When you send an API request, an access log entry is generated. This log entry contains information such as the IP address of the API caller, request URL, response latency, response status code, and number of bytes of the request and response messages. You can use the log data to monitor the operating status of your web services.



Resources

- Projects and Logstores

  > **Note** You must not delete the projects or Logstores that are related to the log management feature of API Gateway. Otherwise, API Gateway access logs cannot be sent to Log Service.

- Dedicated dashboard

  After you enable the feature, a dedicated dashboard is automatically created for API Gateway access logs.

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can customize a dashboard to display query results. For more information, see Create a dashboard.

Dashboard	Description
*Logstore Name*_apigateway_access_logs	Displays the overall log statistics of API Gateway, including the number of requests, request success rate, request error rate, request latency, number of apps that call API operations, request errors, top API groups, top APIs, and top request latencies.

Billing

- You are not charged for using the log management feature of API Gateway.
- After API Gateway access logs are shipped to Log Service, you are charged for the storage space that the log data occupies and the number of read/write operations. You are also charged for reading, transforming, and shipping the data. For more information, see Billing methods.

Limits

- The API Gateway instance and the Log Service project that stores API Gateway access logs must reside in the same region.

- You can specify only one project and Logstore to store API Gateway access logs generated in a region.

4.16.2. Enable the log management feature

This topic describes how to enable the log management feature in the API Gateway console. After you enable the feature, you can use Log Service to collect API Gateway access logs.

Prerequisites

A project and a Logstore are created. For more information, see Create a project and Create a Logstore.

Procedure

> **Note** Before you can use a RAM user to enable the log management feature, you must grant the required permissions to the RAM user. For more information, see RAM user authorization.

1. Log on to the API Gateway console.
2. In the left-side navigation pane, choose **Open API > Logs**.
3. In the top navigation bar, select a region based on your business requirements.
4. On the **Log Management** page, click **Create Log Configuration**.
5. In the **Create Log Configuration** dialog box, select the project and Logstore that you created and click **Confirm**.
6. In the **Tips** dialog box, click **Go to SLS console**.
7. In the Log Service console, enable the indexing feature. For more information, see Configure indexes.

   After you configure indexes for the Logstore, you can query and analyze the logs in the Logstore.

What's next

After Log Service collects API Gateway access logs, you can query, analyze, download, ship, and transform the logs. You can also create alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.16.3. Log fields

This topic describes the fields of API Gateway access logs.

Field	Description
apiGroupUid	The ID of the group to which the API belongs.
apiGroupName	The name of the group to which the API belongs.
apiUid	API ID.
apiName	The name of the API.
apiStageUid	The stage ID of the API.
apiStageName	The stage name of the API.

Field	Description
httpMethod	The HTTP method of the request.
path	The path of the requested resource.
domain	The domain name of the requested resource.
statusCode	The HTTP status code.
errorMessage	The returned error message.
appId	The ID of the client that sends the request.
appName	The name of the client that sends the request.
clientIp	The IP address of the client that sends the request.
exception	The error message returned by the backend server.
providerAliUid	The ID of the Alibaba Cloud account or RAM user that the API belongs.
region	The ID of the region, for example, cn-hangzhou.
requestHandleTime	The time when the request is sent. The time is in GMT.
requestId	The request ID. The ID is globally unique.
requestSize	The size of the request message. Unit: bytes.
responseSize	The size of the response message. Unit: bytes.
serviceLatency	The response latency of the backend server. Unit: milliseconds.

4.17. ActionTrail access logs

4.17.1. Usage notes

ActionTrail provides the operation log shipment feature. You can use this feature to collect operation logs from ActionTrail to Log Service in real time. Log Service allows you to query, ship, transform, and visualize data in real time. You can also create alerts for the data. This way, you can monitor the operations on your cloud resources. This topic describes the resources and billing methods that are related to ActionTrail log shipment.

Resources

- Project

  When you enable the logging feature for ActionTrail, you must specify a destination project.

- Dedicated Logstore

  After you specify a destination project for operations logs, a dedicated Logstore named in the actiontrail_*Trail name* format is automatically created in the project.

  - The indexing feature is automatically enabled for the Logstore. Indexes are configured for some fields.
  - By default, log data is permanently retained in the Logstore. You can modify the retention period of log data based on your business requirements. For more information, see Manage a Logstore.

- Dedicated dashboard

  By default, only one dedicated dashboard is created.

  > **Note** We recommend that you do not make changes to the dedicated dashboard because this may affect the availability of the dashboard. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
actiontrail_*Trail name*_audit_center_cn	Displays the visualized log data of operations on cloud resources in real time. The log data includes page views (PVs), unique views (UVs), the number of source servers, the distribution of event sources, and the trend of the PV/UV ratio.

Billing

- The cost of ActionTrail log management is included in the billing for ActionTrail. For more information, see Billing.
- After SQL audit logs are sent to Log Service, you are charged based on the storage space, read traffic, the number of requests, data transformation, and data shipping. For more information, see Billable items.

Limits

- You can create a trail in the ActionTrail console to ship operation logs to an OSS bucket or a Log Service Logstore. You can create a maximum of five trails in a region.
- You can write only operation logs of cloud services to dedicated Logstores.

Scenarios

- You can use the feature to monitor the operations on the cloud resources of your Alibaba Cloud account and troubleshoot exceptions. You can also use operations logs to track accidental deletions and risky operations.
- You can use log analysis results to track the distribution and sources of operations on essential cloud resources and optimize strategies on the cloud resources.
- You can query operations logs of cloud services such as ActionTrail and view the distribution and time of operations. This way, you can monitor the status of cloud resources in real time.
- You can customize query statements based on your operational and data requirements. You can also customize dashboards to analyze data in real time based on your resource usage and user logons.

4.17.2. Enable the log shipping feature

This topic describes how to enable the log shipping feature in the ActionTrail console. After you enable the log shipping feature, you can ship operations logs of cloud services from ActionTrail to Log Service.

Prerequisites

ActionTrail is authorized to use the AliyunActionTrailDefaultRole role to ship logs to Log Service.

You can go to the Cloud Resource Access Authorization page to complete the authorization.

> **Note**
>
> - This operation is required only when you enable the log shipping feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
> - If you use a RAM user to log on to ActionTrail, you must grant required permissions to the RAM user. For more information, see RAM user authorization.
> - To ensure that operations logs can be shipped to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

Procedure

1. Log on to the ActionTrail console.
2. In the left-side navigation pane, choose **ActionTrail > Trails > Create Trail**.
3. On the **Create Trail** page, set the required parameters.

   For more information about how to set the parameters, see Create a single-account trail.

   i. Configure basic trail settings and click **Next**.

   ii. Configure event delivery settings and click **Next**.

   The following table describes the parameters that you can use to ship operations logs to Log Service.

Parameter	Description
Delivery to Log Service	Select **Delivery to Log Service** to ship logs to Log Service. You can select **New Log Service Project** or **Existing Log Service Project** based on your business requirements.
Logstore Region	The region where the Logstore resides.
Project Name	The name of the project. You can select an existing project or create a project.

   iii. Confirm the configurations and click **Submit**.

What's next

After operations logs of cloud services are shipped from ActionTrail to Log Service, you can query, analyze, download, ship, and transform the logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.17.3. Log fields

This topic describes the fields of operations logs in ActionTrail.

Field	Description
__topic__	The topic of the log entry. Valid value: actiontrail_event.
event	The log event in the JSON format. The content of this field varies depending on the log event.
event.eventId	The unique ID of the event.
event.eventName	The name of the event.
event.eventSource	The source of the event.
event.eventType	The type of the event.
event.eventVersion	The event format version of the event. Valid value: 1.
event.acsRegion	The region where the event occurs.
event.requestId	The ID of the API request.
event.apiVersion	The version of the API.
event.errorMessage	The error message returned if an error occurs during the processing of the API request.
event.serviceName	The name of the Alibaba Cloud service associated with the event, for example, VPC.
event.sourceIpAddress	The IP address from which the request is sent.
event.userAgent	The user agent that sends the API request.
event.requestParameters.HostId	The ID of the server from which resources are requested.
event.requestParameters.Name	The name of the server from which resources are requested.
event.requestParameters.Region	The region where requested resources reside.
event.userIdentity.accessKeyId	The AccessKey ID of the logon account that initiates the API request.
event.userIdentity.accountId	The ID of the logon account that initiates the API request.
event.userIdentity.principalId	The requester ID. For example, if the type parameter is set to ram-user, enter the ID of the logon account.
event.userIdentity.type	The identity type of the logon account. • root-account: Alibaba Cloud account • ram-user: RAM user • assumed-role: RAM role • system: Alibaba Cloud service

Field	Description
event.userIdentity.userName	The requester name. For example, if the type parameter is set to ram-user, enter the name of the logon account.

4.18. Inner-ActionTrail

4.18.1. Usage notes

ActionTrail provides the inner-ActionTrail feature. You can use this feature to ship platform operations logs from ActionTrail to Log Service in real time. Log Service allows you to query, transform, and consume the logs in real time. This topic describes the resources, billings, and limits that are related to the inner-ActionTrail feature.

> **Note** You can use this feature to collect platform operations logs of OSS buckets, ECS instances, RDS instances, Container Service for Kubernetes clusters, and E-MapReduce clusters.

Resources

- Projects and Logstores
  - By default, log data is permanently retained in the Logstore. You can modify the retention period of log data based on your business requirements. For more information, see Manage a Logstore.

    > **Note** You must not delete the projects or Logstores that are related to platform operations logs. Otherwise, the platform operations logs cannot be sent to Log Service.

- Dedicated dashboard

  After you enable the feature, a dedicated dashboard is automatically created for the platform operations logs.

  > **Note** We recommend that you do not make changes to the dedicated dashboard because this may affect the usability of the dashboard. You can customize a dashboard to display query results. For more information, see Create a dashboard.

Dashboard	Description
actiontrail_ *Trail Name*_audit_center_en	Displays the visualized log data of operations on cloud resources in real time. The log data includes page views (PVs), unique views (UVs), the number of source servers, distribution of event sources, and trend of the PV/UV ratio.

Billing

- You are not charged for using the inner-AcionTrail feature of ActionTrail.
- After platform operations logs are shipped to Log Service, you are charged for the storage space that the log data occupies and the number of read/write operations. You are also charged for reading, transforming, and shipping the data. For more information, see Billing methods.

Limits

- To use the inner-ActionTrail feature, you must be granted relevant permissions. To do so, submit a ticket or contact technique support.
- You must clear overdue payments to make sure that Log Service is available.
- All platform operations logs are shipped to the same Logstore.
- You can write only platform operations logs to a dedicated Logstore.
- You cannot modify the retention period for the platform operations logs stored in a dedicated Logstore.

Benefits

- Classified protection compliance: stores the platform operations logs of Alibaba Cloud services for six months or more to meet the classified protection compliance requirements.
- Ease of use: allows you to collect platform operations logs in real time with simple configurations.
- Real-time analysis: provides real-time analysis capabilities and out-of-the-box dashboards. This allows you to monitor the distribution and other details of platform operations logs.
- Real-time alerts: supports real-time monitoring and alerting based on customized metrics. This ensures that you can respond to critical business exceptions at the earliest opportunity.
- Collaboration: allows you to integrate the feature with stream computing, cloud storage, visualization, and other data ecosystems to perform finer-grained analysis.

Scenarios

- Track platform operations logs of Alibaba Cloud services and locate the causes of resource changes.
- View, audit, and evaluate platform operations logs in near real time.
- Export platform operations logs to data centers.

4.18.2. Enable the inner-ActionTrail feature

This topic describes how to enable the inner-ActionTrail feature and collect operations logs of cloud services by using the Log Service console.

Prerequisites

- You are authorized to use the inner-ActionTrail feature.
- ActionTrail is authorized to ues the AliyunActionTrailDefaultRole role to ship logs to Log Service.

  You can go to the Cloud Resource Access Authorization page to complete the authorization.

  > **Note**
  > - This operation is required only when you enable the inner-ActionTrail feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
  > - If you use a RAM user to log on to ActionTrail, you must grant required permissions to the RAM user. For more information, see RAM user authorization.
  > - To ensure that operations logs can be shipped to Log Service, do not revoke permissions from the RAM role or delete the RAM role.

- A project and a Logstore are created. For more information, see Quick start.

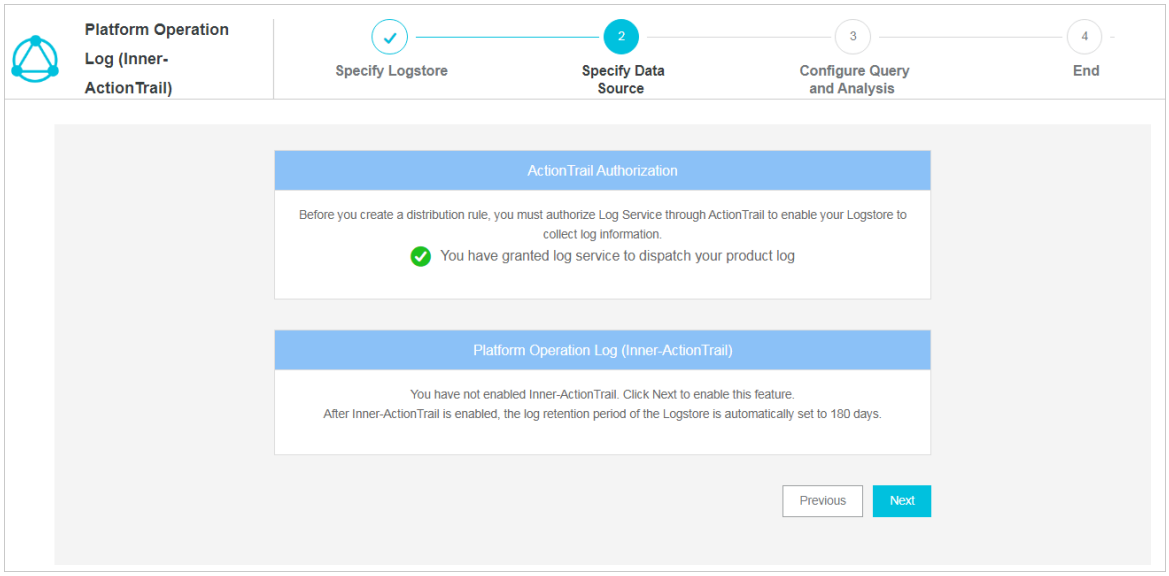
Procedure

1.
2. In the **Import Data** section, click the **Platform Operation Log (Inner-ActionTrail)** card.

   You can also log on to the ActionTrail console. On the page that appears, choose **Inner-ActionTrail > Trails**. On the Create Trail page, set the parameters to ship operations logs to Log Service.

   > **Note**
   > - If you enable the inner-ActionTrail feature in the ActionTrail console, Log Service creates a dedicated Logstore named innertrail_Trail Name.
   > - If you enable the inner-ActionTrail feature in the Log Service console, the settings that you configure in the Log Service console are not synchronized to the ActionTrail console. If you enable the inner-ActionTrail feature in the Log Service console and create a trail in the ActionTrail console, the settings that you configure in the ActionTrail console overwrite the settings that you configure in the Log Service console.
   > - If you cannot find the **Platform Operation Log (Inner-ActionTrail)** card in the **Import Data** section or **Inner-ActionTrail > Trails** in the ActionTrail console, submit a ticket or contact technical support.

3. Select the project and Logstore. Then, click **Next**.
4. In the **Specify Data Source** step, click **Next**.



> **Note**
>
> - All operations logs are shipped to only one Logstore.
> - You can disable the inner-ActionTrail feature in the **Specify Data Source** step.
> - You can disable the inner-ActionTrail feature by choosing **Inner-ActionTrail > Trails** in the ActionTrail console.
> - After you disable the inner-ActionTrail feature, new logs are not shipped to the dedicated Logstore. The logs that have been shipped to the dedicated Logstore are automatically deleted after the retention period expires.

5. In the **Configure Query and Analysis** step, click **Next**.

   The indexing feature is automatically enabled for the dedicated Logstore and indexes are created for the data in the Logstore.

What's next

After operations logs are shipped to Log Service, you can query, analyze, download, ship, and transform the logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.18.3. Fields in Alibaba Cloud-initiated events

This topic describes the fields in Alibaba Cloud-initiated events and provides an example of such an event.

Fields that are contained

Field	Description
EventID	The ID of the event. This field uniquely identifies an event.
EventVersion	The version of the event.
EventProduct	The name of the Alibaba Cloud service on which an operation is performed, such as Object Storage Service (OSS).
EventName	The name of the event that occurs when you call an API operation on an Alibaba Cloud service, such as Set Bucket Quota Limit.
EventDescription	The reason why the operation is performed. For example, this field records the ID of a regular ticket, an internal operations and maintenance (O&M) ticket, a change ticket, or a security scanning task.

Field	Description
EventType	The type of the operation. Valid values: • CUSTOMER_INITIATED_SUPPORT Indicates a technical support operation that is performed by Alibaba Cloud engineers, such as troubleshooting based on tickets that are submitted by users. • ALIYUN_INITIATED_SERVICE Indicates an operation that is performed by Alibaba Cloud engineers or systems based on O&M requirements, such as bucket migration across clusters after the cluster hardware is out of warranty. • ALIYUN_INITIATED_PENALTY Indicates an operation that is performed by Alibaba Cloud engineers or systems on the public data of users based on rules and regulations.
EmployeeID	The encrypted ciphertext of the globally unique employee ID of the operator in Alibaba Cloud. If an event is automatically operated by a system program, this field is empty. If an event is manually operated by an Alibaba Cloud engineer, this field is not empty. When necessary, you can submit a ticket and provide the value of this field to Alibaba Cloud to query the specific operator of an event.
EventMethod	The method that the operator uses to perform the operation. For example, the operator uses the API, SDKs, or console of a specific Alibaba Cloud service to read, write, back up, or restore data.
ResourceType	The type of the resource that is associated with the event. Example: ACS::ACK::Cluster.
ResourceID	The ID of the event-associated resource, such as the ID of an OSS bucket.
ResourceRegionID	The ID of the region where the event-associated resource resides.
ResourceOwnerID	The ID of the Alibaba Cloud account to which the event-associated resource belongs.
EventAdditionalDetail	The additional information about the event.
EventTime	The time when the operation is performed. The time is in UTC. Example: 2021-03-22T05:23:37Z.
EventLevel	The severity level of the operation. Valid values: • NOTICE: indicates that the system only records the operation in the log of an Alibaba Cloud-initiated event. • WARNING: indicates that the system records the operation in the log of an Alibaba Cloud-initiated event and sends an alert notification to users.

Field	Description
EventLocation	The geographic location of the country where the Alibaba Cloud engineer performs the operation. Example: CN. The value indicates mainland China.

Examples


```
{
    "EmployeeID": "64tSfLheCbLra9ClKaUF86J4DkP84p3n6H6sc4BS****",
    "EventAdditionalDetail": "{\"filter\":\"user_id:153915067560****\",\"groupbys\":\"ts,storage_type\",\"max\":\"100000\",\"endts\":\"1616947199\",\"orderby\":\"ts\"}",
    "EventDescription": "requestID: 61167C65-B80D-4876-A573-D61DD4238AA2",
    "EventID": "4facb9c7-d970-4f53-af5b-4ee08f51****",
    "EventLevel": "NOTICE",
    "EventLocation": "CN",
    "EventMethod": "Regular Read",
    "EventName": "DescribeK8sResourceGroup",
    "EventProduct": "ACK",
    "EventTime": "2021-03-29T09:44:51Z",
    "EventType": "ALIYUN_INITIATED_SERVICE",
    "EventVersion": "1.0.0",
    "ResourceID": "cd63fb222a3be44a89df72686b343****",
    "ResourceOwnerID": "129242164613****",
    "ResourceRegionID": "cn-hangzhou",
    "ResourceType": "ACS::ACK::Cluster"
}
```


4.19. PolarDB-X 1.0 SQL audit logs

4.19.1. Usage notes

The SQL audit and analysis feature of PolarDB-X 1.0 allows you to collect SQL audit logs and send the logs to Log Service. You can query, ship, and transform the collected logs. You can also visualize log analysis results and configure alerts for the logs. This topic describes the assets, billing, and limits of the SQL audit and analysis feature in PolarDB-X 1.0.

Assets

- Dedicated projects and Logstores

  After you enable the SQL audit and analysis feature, Log Service creates a project named drds-audit-region name-Alibaba Cloud account ID and a Logstore named drds-audit-log.

- Dedicated dashboards

  After you enable the SQL audit and analysis feature, Log Service generates three dashboards by default.

> **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
Operation Center (Simple)	Shows the statistics of DRDS instances, including the metrics, distribution, and trends of SQL statement executions.
Performance Center	Shows the performance metrics, average time consumed by each type of SQL statement, and the distribution and sources of slow SQL queries.
Security Center	Shows the security metrics, batch delete events, and malicious SQL executions.

Billing

- You are not charged for the SQL audit and analysis feature in the PolarDB-X 1.0 console.
- After SQL audit logs are sent to Log Service, you are charged by Log Service based on the storage space, read traffic, the number of requests, data transformation, and data shipping. For more information, see Log Service pricing.

Limits

- You can write only PolarDB-X 1.0 SQL audit logs to a dedicated Logstore. In addition, you cannot modify the indexes in a dedicated Logstore.
- The SQL audit and analysis feature of PolarDB-X 1.0 is available in the following regions: China (Hong Kong) and Singapore (Singapore).

  The Log Audit Service application of Log Service also supports access to PolarDB-X 1.0 SQL audit logs. For more information, see Log Audit Service.

4.19.2. Enable the SQL audit and analysis feature

This topic describes how to enable the SQL audit and analysis feature in the PolarDB-X 1.0 console.

Prerequisites

- Log Service is activated.
- A PolarDB-X 1.0 instance is purchased.
- A database is created.

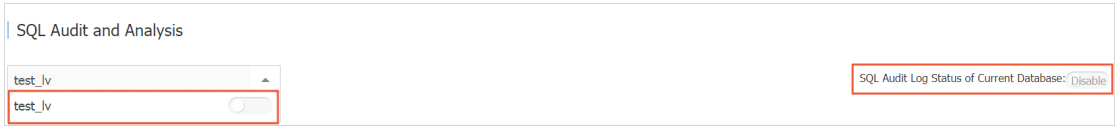
Procedure

1. Log on to the PolarDB-X 1.0 console.
2. In the upper-left corner of the page, select the region where the instance resides.
3. On the **Instance List** page, click the name of the instance.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > SQL Audit and Analysis**.

5. Authorize PolarDB-X 1.0 as prompted to assume the AliyunDRDSDefaultRole role to access Log Service.

> **Note**
> - This operation is required only when you enable the SQL audit and analysis feature for the first time. You must complete the authorization by using your Alibaba Cloud account.
> - If you use a RAM user to log on to PolarDB-X 1.0, you must grant required permissions to the RAM user. For more information, see RAM user authorization.
> - To ensure that PolarDB-X 1.0 SQL audit logs can be sent to Log Service, do not delete the RAM role.

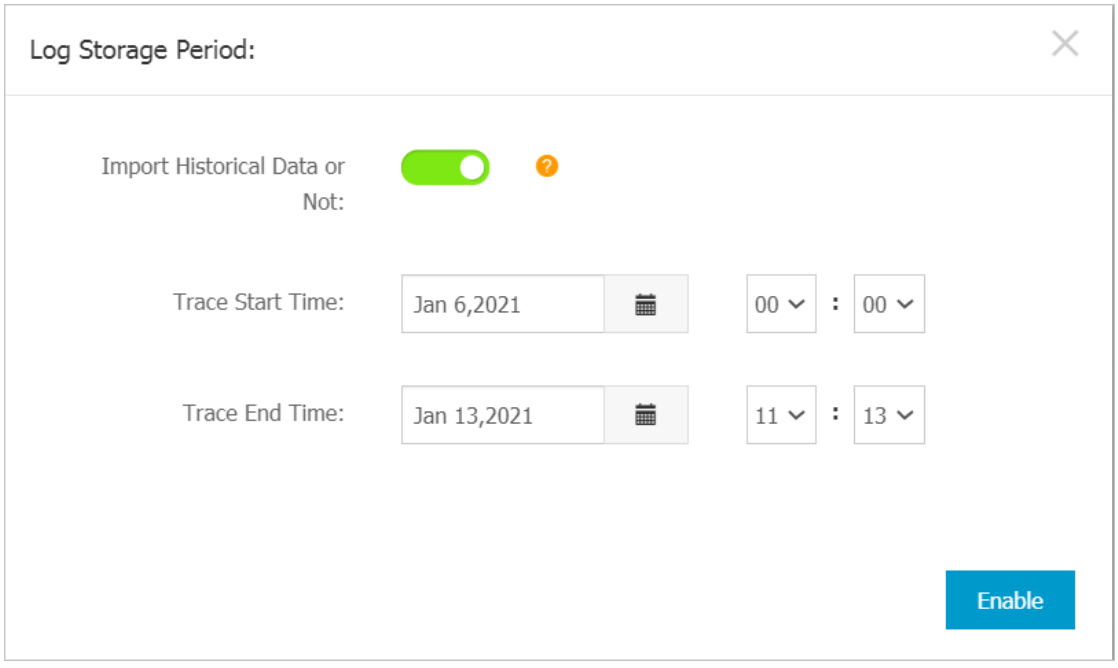
6. Enable the SQL audit and analysis feature.
   i. Select the database, and then turn on the switch next to the database or turn on the SQL Audit Log Status of Current Database switch.



   ii. (Optional)Import historical data.

   By default, only the logs that are generated after the SQL audit and analysis feature is enabled are sent to Log Service. If you want to analyze logs that are generated before the SQL audit and analysis feature is enabled, you can import historical data. To do so, you can turn on the **Import Historical Data or Not** switch in the **Log Storage Period** dialog box. Then, set the **Trace Start Time** and **Trace End Time** parameters. You can import up to seven days of historical data.

   You can view the log import progress in the **Task Progress** dialog box in the upper-right corner of the current page.



iii. Click **Enable**.

Result

After PolarDB-X 1.0 SQL audit logs are collected and sent to Log Service, you can query, analyze, download, ship, and transform the collected logs. You can also configure alerts for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.19.3. Log fields

This topic describes the fields of PolarDB-X 1.0 SQL audit log entries.

Log field	Description
__topic__	The topic of a log entry. The value must be in the format of drds_audit_log_*instance ID_database name*, for example, drds_audit_log_drdsxyzabcd_demo_drds_db.
affect_rows	The number of rows that are returned after an SQL statement is executed. • The number of rows that are affected when an SQL statement is executed to add, delete, or modify data. • The number of rows that are returned after a query statement is executed. This field is available for instance version 5.3.4-15378085 and later.
client_ip	The IP address of a client that accesses a PolarDB-X 1.0 instance.
client_port	The port number of a client that accesses a PolarDB-X 1.0 instance.
db_name	The name of a PolarDB-X 1.0 database.
fail	The result of an SQL statement. Valid values: • 0: The SQL statement is executed. • 1: The SQL statement fails to be executed. This field is available for instance version 5.3.4-15378085 and later.
hint	The hint that is used to execute an SQL statement.
instance_id	The ID of a PolarDB-X 1.0 instance.
response_time	The response time. Unit: milliseconds. This field is available for instance version 5.3.4-15378085 and later.
sql	The SQL statement.
sql_code	The hash value of a template SQL statement.
sql_time	The time when an SQL statement is executed. The value is in the format of yyyy-MM-dd HH:mm:ss.SSS.

Log field	Description
sql_type	The type of an SQL statement. Valid values: Select, Insert, Update, Delete, Set, Alter, Create, Drop, Truncate, Replace, and Other.
sql_type_detail	The name of an SQL parser.
table_name	The name of a database table. Multiple tables are separated by commas (,).
trace_id	The trace ID of an SQL statement when the statement is executed. If a transaction is executed, it is tracked by using an ID. The ID consists of a trace ID, a hyphen (-), and a number, for example, drdsabcdxyz-1 and drdsabcdxyz-2.
user	The name of the user who executes an SQL statement.

4.20. RDS SQL execution logs

4.20.1. Usage notes

Log Service and ApsaraDB RDS jointly launch the feature that allows you to ship the SQL audit logs of ApsaraDB RDS databases to Log Service. Log Service allows you to perform various operations. For example, you can query data in real time, analyze data in visualized mode, ship data, transform data, and configure alerts. This topic describes the assets, billing, and limits of the feature.

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

- After you enable the SQL Explorer feature for an ApsaraDB RDS for MySQL instance, you are charged for the feature on an hourly basis. The hourly fee is calculated by using the following formula: Hourly fee = Amount of audit log data per hour × Unit price.

  > **Note** If your ApsaraDB RDS for MySQL instance runs the RDS Enterprise Edition, you can use the SQL Explorer feature free of charge.

- After logs are shipped to Log Service, you are charged for the storage space occupied by the logs, read traffic, number of requests, data transformation, and data shipping. For more information, see Pay-as-you-go.

Limits

- You can ship the SQL audit logs only from the following type of ApsaraDB RDS instance to Log Service:

  ApsaraDB RDS for MySQL instances: All available RDS editions except the RDS Basic Edition are supported.

- You can ship SQL audit logs from an Apsara RDS instances to Log Service only after you enable the SQL Explorer feature for the instance.
- The ApsaraDB RDS instance from which you want to ship SQL audit logs to Log Service must reside in the same region as the project to which you want to ship the logs.
- All regions except Local Regions are supported.

4.20.2. Collect RDS SQL audit logs

This topic describes how to collect RDS SQL audit logs by using the Log Service console.

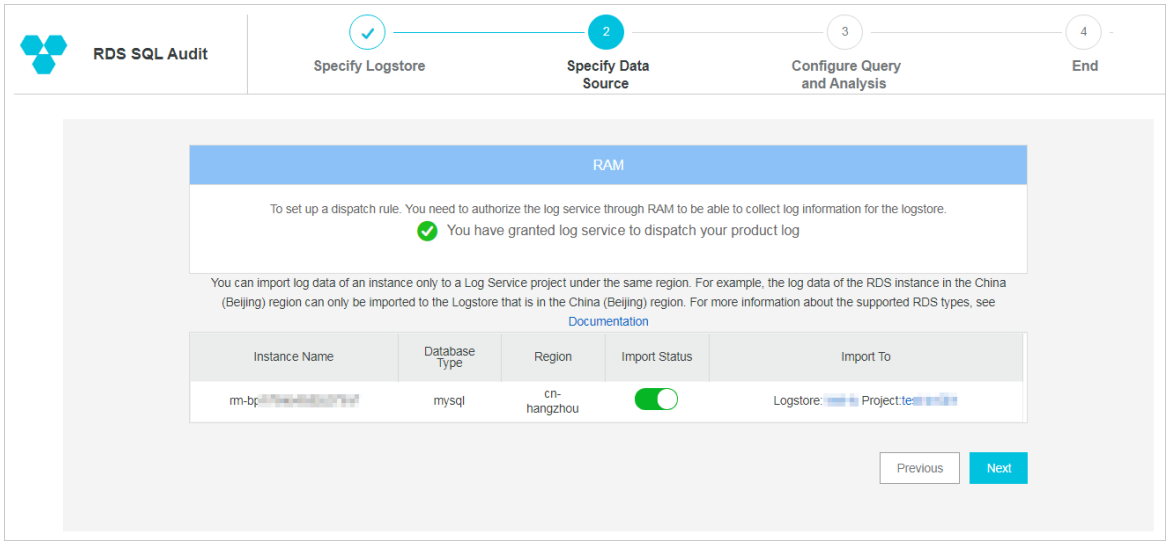
Prerequisites

- An ApsaraDB RDS instance is created. If an ApsaraDB RDS for MySQL instance is created, the SQL Explorer feature of a paid edition is enabled for the instance. For more information, see Create an ApsaraDB RDS for MySQL instance and Use the SQL Explorer feature on an ApsaraDB RDS for MySQL instance.
- A Log Service project and Logstore are created in the region where the RDS instance resides. For more information, see Create a project and a Logstore.

Procedure

1.
2. In the **Import Data** section, select **RDS SQL Audit - Cloud Products**.
3. In the **Specify Logstore** step, select a project and Logstore, and click **Next**.
4. In the **Specify Data Source** step, complete RAM authorization, enable the data shipping feature, and then click **Next**.

   > **Note**
   > - If you have not authorized Log Service to ship logs, click **Authorize** next to **RAM**, and complete the authorization as prompted.
   > - The destination ApsaraDB RDS instance may not appear on the prompted page or you may fail to enable the data shipping feature. This issue occurs when your ApsaraDB RDS instance does not meet the required conditions. For more information about how to check whether your ApsaraDB RDS instance meets the required conditions, see the Prerequisites section.



5. In the **Configure Query and Analysis** step, click **Next**.

   By default, the indexing feature is enabled for the Logstore where RDS SQL audit logs are stored. Indexes are configured for these audit logs. You can modify indexes as needed. For more information, see Configure indexes.

What's next

After RDS SQL audit logs are collected by Log Service, you can search, analyze, download, ship, and transform these logs. You can also configure alerts for these logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.20.3. 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.
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.

Log field	Description
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.

4.21. ApsaraDB for Redis logs

4.21.1. Usage notes

The log audit feature of ApsaraDB for Redis allows you to collect and send audit logs, slow query logs, and operational logs to Log Service. You can query, ship, and transform the collected logs. You can also visualize log analysis results and configure alerts for the logs. This topic describes the assets, billing, and limits of the log audit feature in ApsaraDB for Redis.

Assets

- Dedicated project and Logstores

  After you enable the log audit feature of ApsaraDB for Redis, Log Service creates a project named nosql-Alibaba Cloud account ID-region ID and Logstores named redis_audit_log and redis_slow_run_log.

  - The redis_audit_log Logstore is used to store the audit logs of ApsaraDB for Redis.
  - The redis_slow_log Logstore is used to store the slow query logs and operational logs of ApsaraDB for Redis.

- Dedicated dashboards

  > **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can customize a dashboard to display query results. For more information, see Create a dashboard.

  - After you enable the log audit feature, Log Service generates a dashboard for the redis_audit_log Logstore by default.

Dashboard	Description
Redis Audit Center	Shows the statistics of the audit logs of ApsaraDB for Redis. The information displayed on the dashboard includes the number of users, clients, and audit log entries, average response time (RT), and average queries per second (QPS).

- After you enable the log audit feature, Log Service generates a dashboard for the redis_slow_run_log Logstore by default.

Dashboard	Description
Redis Slow Log Center	Shows the statistics of the slow query logs of ApsaraDB for Redis. The information displayed on the dashboard includes the number of users, clients, and audit log entries, average RT, and average QPS.

Billing

- You are charged for ApsaraDB for Redis instances based on the storage space and data retention period. The billing varies by region. For more information, see Billable items and prices.

  You are not charged when you write, store, and query slow query logs and operational logs.
- After logs are pushed from ApsaraDB for Redis to Log Service, you are charged for data transformation and log shipping. You are also charged for the read traffic over the Internet. For more information, see Log Service pricing.

Limits

- You can write only the logs of ApsaraDB for Redis to a dedicated Logstore. You cannot modify the attributes of a dedicated Logstore. In addition, you cannot modify or delete the indexes of a dedicated Logstore.
- You cannot modify the data retention period of a dedicated Logstore by using the Log Service console, API, or SDK. You can modify the data retention period of a dedicated Logstore in the ApsaraDB for Redis console.
- If you have overdue payments for your Log Service resources, the log analysis feature is automatically stopped. To ensure service continuity, you must pay your overdue payment within the prescribed time limit.
- The log audit feature is available for only ApsaraDB for Redis instances whose version is Redis 4.0 or later and minor version is the latest. The editions of ApsaraDB for Redis include Community Edition and Enhanced Edition.

4.21.2. Enable the log audit feature

This topic describes how to enable the log audit feature in the ApsaraDB for Redis console. This topic also describes how to use the log audit feature to push Redis audit logs to Log Service.

Prerequisites

An ApsaraDB for Redis instance that meets the following conditions is created:

- The instance is an ApsaraDB for Redis Community Edition instance or a performance-enhanced instance of the ApsaraDB for Redis Enhanced Edition (Tair). For more information about performance-enhanced instances, see Performance-enhanced instances.
- The engine version of the instance is Redis 4.0 or later, and the latest minor version is used. For more information about how to update the minor version and upgrade the engine version of an instance, see Upgrade the major version and Update the minor version.

Procedure

> **Note** If you use a RAM user to enable the log audit feature, you must grant the required permissions to the RAM user. For more information, see RAM user authorization.

1.
2. In the left-side navigation pane, click **Instances**.
3. In the top navigation bar, select the resource group and region to which the ApsaraDB for Redis instance belongs.
4. On the **Instances** page, click the ApsaraDB for Redis instance.
5. In the left-side navigation pane, choose **Logs > Audit Log**.
6. Select **Official Edition**, set the **Log Retention Period** parameter, and then click **Estimate Fees and Enable Audit Logs**.
7. In the **Estimate Fees and Enable Audit Logs** dialog box, set the **Average Writes per Second** and **Average Data Size per Write** parameters. Then, click **Enable**.

What's next

After Log Service collects ApsaraDB for Redis logs, you can query, analyze, download, ship, and transform the logs. You can also create alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.21.3. Log fields

Redis logs include audit logs, slow logs, and operational logs. This topic describes the fields of these logs.

Audit logs

The audit logs are stored in the Logstore named redis_audit_log. The following table describes the log fields.

Field	Description
__topic__	The topic of the log entry. • redis_audit_log: audit logs of the database • redis_proxy_audit_log: audit logs of the proxy
account	The name of the database account.
command	The Redis command that is run on the database.
db	The name of the database.
extend_information	The additional information.
instanceid	The ID of the Redis instance.
ip	The IP address of the database server.

Field	Description
is_cautious	Indicates whether the operation is dangerous. Valid values: • 0: No • 1: Yes
latency	The request latency.
time	The timestamp, for example, 1597048424.
type	The type of the log entry.

Slow logs

The slow logs are stored in the Logstore named redis_slow_run_log. The following table describes the log fields.

Field	Description
__topic__	The topic of the log entry. • redis_audit_log: audit logs of the database • redis_proxy_audit_log: audit logs of the proxy
account	The name of the database account.
command	The Redis command that is run on the database.
db	The name of the database.
extend_information	The additional information.
instanceid	The ID of the Redis instance.
ip	The IP address of the database server.
is_cautious	Indicates whether the operation is dangerous. Valid values: • 0: No • 1: Yes
latency	The request latency.
time	The timestamp, for example, 1597048424.
type	The type of the log entry.

Operational logs

The operational logs are stored in the Logstore named redis_slow_run_log. The following table describes the log fields.

Field	Description
__topic__	The topic of the log entry. Valid value: redis_run_log.
extend_information	The additional information.
instanceid	The ID of the Redis instance.
node_type	The type of the log entry. • proxy: operational logs of the proxy • db: operational logs of the database
runlog	The content of the operational log entry.
time	The timestamp, for example, 1597048424.

4.22. MongoDB logs

4.22.1. Usage notes

ApsaraDB for MongoDB provides the log audit feature that is supported by Log Service. You can use this feature to query audit logs, slow logs, and operational logs of ApsaraDB for MongoDB and visualize the query results. You can also create alert rules for logs, ship logs, and transform logs. This topic describes the resources, billing method, and limits that are related to the log audit feature.

Assets

- Dedicated project and Logstores

  If you enable the log audit feature of ApsaraDB for MongoDB, a project and two Logstores are created. The project is named in the nosql-Alibaba Cloud account ID-region ID format and the Logstores are named mongo_audit_log and mongo_slow_run_log.

  - The mongo_audit_log Logstore is used to store ApsaraDB for MongoDB audit logs.
  - The mongo_slow_run_log Logstore is used to store slow logs and operational logs of ApsaraDB for MongoDB.

- Dedicated dashboards

  > **Note** Changes to dedicated dashboards may affect the usability of the dashboards. We recommend that you do not make changes to dedicated dashboards. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

  A dashboard is automatically generated for the mongo_audit_log Logstore.

Dashboard	Description
Mongo Audit Log Center	Displays audit logs of ApsaraDB for MongoDB. The log data includes the number of users, the number of clients, the average response time (RT), and the average request rate.

Billing

- The log audit feature of ApsaraDB for MongoDB includes the trial version and official version. The trial version is available and is free of charge.
- If you want to query, analyze, or create alerts for logs that are pushed to Log Service, you are charged for indexes and alert notifications in Log Service. For more information, see Log Service pricing.

> **Note** After audit logs are pushed to Log Service, you cannot pull, consume, ship, or transform the audit logs.

Limits

- You can write only ApsaraDB for MongoDB logs to a dedicated Logstore. You cannot modify the indexes in a dedicated Logstore.
- You cannot delete dedicated projects or Logstores.
- The retention period of data in a dedicated Logstore is one day and cannot be changed.
- If you have overdue payments for your Log Service resources, the log analysis feature becomes unavailable. To ensure service continuity, you must settle your overdue payment within the specified time limit.
- The log audit feature is available for ApsaraDB for MongoDB replica set instances and sharded cluster instances.

4.22.2. Enable the log audit feature

This topic describes how to enable the log audit feature in the ApsaraDB for MongoDB console and send audit logs to Log Service.

Prerequisites

A replica set instance with three or more nodes is created, or a sharded cluster instance is created. For more information, see Create a replica set instance and Create a sharded cluster instance.

Procedure

1.
2.
3. In the top navigation bar, select the resource group and region of your instance.
4. In the instance list, click the instance.
5. In the left-side navigation pane, choose **Data Security > Audit Logs**.
6. If this is your first time to use the log audit feature, follow the on-screen instructions to complete authorization.

   After the authorization is complete, the system generates the AliyunServiceRoleForMongoDB RAM role. Your instance can assume the AliyunServiceRoleForMongoDB RAM role to access Log Service resources within your Alibaba Cloud account. For more information, see ApsaraDB for MongoDB service-linked roles.

> **Notice** Do not revoke the permissions from the RAM role or delete the RAM role. If you revoke the permissions from the RAM role or delete the RAM role, the audit logs of the ApsaraDB for MongoDB instance cannot be sent to Log Service.

7. On the **Latest Audit Logs** page, specify the log retention period and click **Enable Audit Logs**.
8. In the **Enable Audit Logs** message, click **OK**.

What's next

After the logs of your instance are collected to Log Service, you can query, analyze, download, ship, and transform the logs. You can also configure alerts based on the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.22.3. Log fields

MongoDB logs include audit logs, slow logs, and operational logs. This topic describes the fields of these logs.

Audit logs

The audit logs are stored in the Logstore named mongo_audit_log. The following table describes the log fields.

> **Note** The names of the fields in audit log and slow logs are the same. You can distinguish the two types of logs based on the value of the audit_type field. The value of the audit_type field in slow logs is *slowop*. If the value of this field in a log entry is not slowop, this log entry is an audit log entry.

Field	Description
__topic__	The topic of the log entry. Valid value: mongo_audit_log.
audit_type	The type of the log entry, for example, Command.
coll	The dataset.
db	The name of the database.
docs_examined	The number of scanned rows.
instanceid	The ID of the ApsaraDB for MongoDB instance.
keys_examined	The number of rows of scanned indexes.
latency	The response latency.

Field	Description
optype	The type of the operation. Valid values: • query: query data • find: search for data • insert: insert data • update: update data • delete: delete data • remove: remove data • getMore: read data • command: operation command
return_num	The number of entries returned.
thread_id	The ID of the thread.
time	The timestamp.
user	The account that is used to log on to the ApsaraDB for MongoDB database.
user_ip	The IP address of the client used to access the ApsaraDB for MongoDB instance.

Slow query logs

The slow logs are stored in the Logstore named mongo_slow_run_log. The following table describes the log fields.

Field	Description
__topic__	The topic of the log entry. Valid value: mongo_run_log.
audit_type	The type of the log entry. Valid value: slowop.
coll	The dataset.
db	The name of the database.
docs_examined	The number of scanned rows.
instanceid	The ID of the ApsaraDB for MongoDB instance.
keys_examined	The number of scanned rows.
latency	The response latency.

Field	Description
optype	The type of the operation. Valid values: • query: query data • find: search for data • insert: insert data • update: update data • delete: delete data • remove: remove data • getMore: read data • command: operation command
return_num	The number of entries returned.
thread_id	The ID of the thread.
time	The timestamp.
user	The account that is used to log on to the ApsaraDB for MongoDB database.
user_ip	The IP address of the client used to access the ApsaraDB for MongoDB instance.

Operational logs

The operational logs are stored in the Logstore named mongo_slow_run_log. The following table describes the log fields.

Field	Description
__topic__	The topic of the log entry. Valid value: mongo_run_log.
category	The type of the log entry, for example, NETWORK logs.
connection	The log connection information.
content	The log content.
instanceid	The ID of the ApsaraDB for MongoDB instance.
ip	The IP address of the database server.
level	The severity of the log entry.
port	The port number.
time	The time when the log entry was generated.

4.23. IoT Platform logs

4.23.1. Usage notes

Alibaba Cloud IoT Platform provides the log dump feature. You can use this feature to dump logs from IoT Platform to Log Service. You can query, ship, and transform logs in real time in the Log Service console. You can also visualize the results of log analysis and configure alerts. This topic describes the resources, billing, and limits of IoT Platform logs.

Resources

- Dedicated projects and Logstores

  After you enable the log dump feature, IoT Platform creates a project named iot-log-Alibaba Cloud account ID-Region ID and a Logstore named iot_logs in the corresponding region.

- Dedicated dashboards

  After you enable the log dump feature, a dashboard is generated by default.

  > **Note** The dedicated dashboard may be updated at any time. We recommend that you do not modify the dedicated dashboard. You can manually create a dashboard to visualize the results of log analysis. For more information, see Create a dashboard.

Dashboard	Description
IoT operation center	Shows the statuses and errors of devices that are connected to IoT Platform. You can view the number of times that you have logged on and off the devices, device IP addresses distributed by region, and error distribution for data parsing script. You can also view the error distribution for Thing Specification Language (TSL) validation, the number of forwarded subscription messages at the server side, the number of forwarded cloud service messages, and error distribution for API calls.

Billing

- You are not billed when you use the log dump feature to dump logs.
- However, you are billed based on the storage space usage, read traffic, number of requests, and the amount of transformed and shipped data. For more information, see Log Service pricing.

Limits

- You can write only IoT Platform logs to a dedicated Logstore. You cannot modify the indexes of the Logstore.
- You cannot delete dedicated projects or Logstores.
- You can create only one log configuration file in each region. The operational logs of all cloud services in the region are automatically dumped to the dedicated Logstore based on the log collection configurations in the file.
- The service is available only in the following regions: China (Shanghai), China (Shenzhen), Singapore (Singapore), Japan (Tokyo), Germany (Frankfurt), US (Silicon Valley), and US (Virginia).

4.23.2. Enable the log dump feature

This topic describes how to enable the log dump feature in the IoT Platform console to dump operational logs to Log Service.

Prerequisites

- Log Service is activated.
- A product is created and a device is connected to the product. For more information, see Connect a device to a product.

Procedure

1. Log on to the IoT Platform console.

   > **Note** The log dump feature is supported only by Alibaba Cloud accounts.

2. In the left-side navigation pane, choose **Maintenance > Device Log**.
3. On the Device Log page, select the product and click the **Log Dump** tab.
4. On the Log Dump tab, click **Enable** to enable the log dump feature.
5. In the **Log Configurations** dialog box, click **OK**.

   After the log dump feature is enabled, the system creates a service linked role named AliyunServiceRoleForIoTLogExport. IoT Platform uses this role to access your cloud resources in Log Service.

   > **Note**
   > - If the AliyunServiceRoleForIoTLogExport role already exists, it cannot be created again.
   > - Do not cancel the RAM role authorization or delete the RAM role. Otherwise, the operational logs of cloud resources in IoT Platform cannot be dumped to Log Service.

What's next

- After you dump logs from IoT Platform to Log Service, you can query, analyze, download, ship, and transform log data. You can also configure alerts for log data. For more information, see Common operations on logs of Alibaba Cloud services.
- If you no longer need to dump logs to Log Service, click **Stop Dump** on the Log Dump tab.

  After you disable the log dump feature, newly generated logs are no longer exported to Log Service. Exported logs will be automatically cleared from Log Service when the specified time for log storage is reached.

4.23.3. Log fields

This topic describes the fields of IoT Platform logs.

Log field	Description
__topic__	The topic of a log. Valid value: iot_log.
bizCode	The type of business.

Log field	Description
code	The status code of a request. • 200: indicates a successful request. • Other status codes indicate a failed request. For more information, see IoT Platform logs.
content	The content of a log.
deviceName	The name of the device that is connected to IoT Platform.
instanceId	The ID of an instance.
messageId	The ID of a message.

operation	The operation that is performed by a device. Operations can be divided into the following types:
	• Firmware update:
	  ◦ OTAFirmwarePush: pushes notifications when a firmware update is initiated, confirmed, and released.
	  ◦ OTAVersionReport: reports the firmware version of a device.
	  ◦ OTAProgressReport: reports the update progress of a device.
	• Data parsing:
	  ◦ RawDataToProtocol: converts raw data to Alink-based data.
	  ◦ ProtocolToRawData: converts Alink-based data to raw data.
	• TSL data submission:
	  ◦ *check*: checks the TSL.
	  ◦ For more information about the method parameter in the message body, see What is a TSL model?.
	• Device behavior management:
	  ◦ online: connects a device to IoT Platform.
	  ◦ offline: disconnects a device from IoT Platform.

productKey	The key of a product. All products share a Logstore. The ProductKey field is used to identify products.
reason	The error cause.
requestId	The ID of a request.
status	The result of a request. • true: indicates a successful request. • false: indicates a failed request.
utcTime	The collection time. The time is in UTC.

Log field	Description
traceId	The trace ID of a request.
clientId	The ID of the client reported to the device.
params	The input parameters.
resultData	The result of a request.

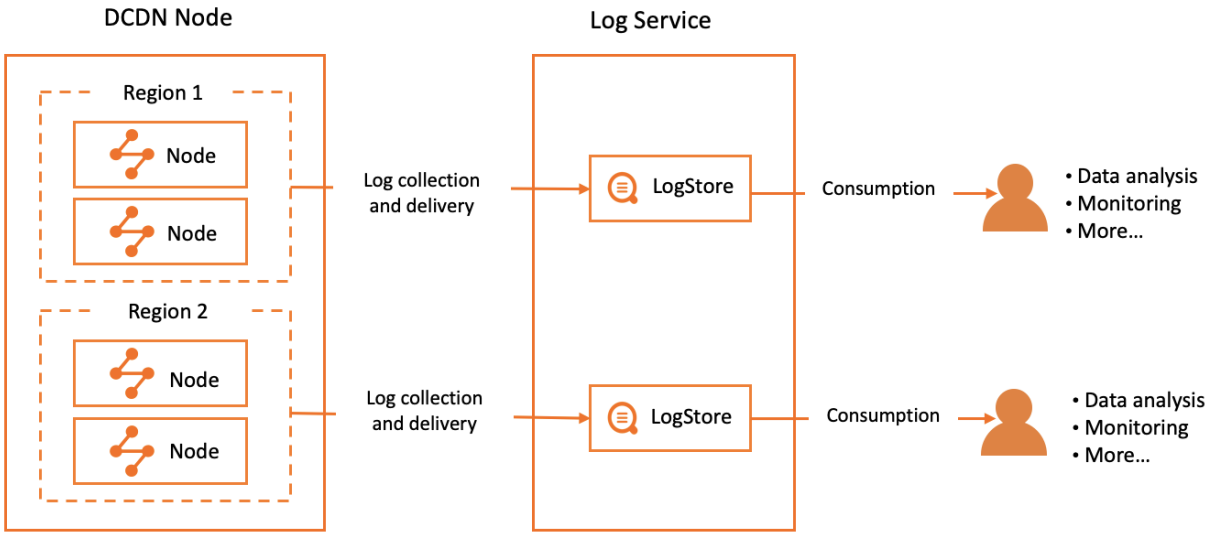
4.24. DCDN real-time logs

4.24.1. Usage notes

The real-time log delivery feature of Dynamic Route for CDN (DCDN) allows you to collect DCDN real-time logs by using Log Service in near real time. After the logs are delivered to Log Service, you can query, ship, and transform the logs in real time in the Log Service console. You can also analyze log data, visualize analysis results, and configure alert rules for logs. This topic describes the assets, billing, and limits of the real-time log delivery feature in DCDN.

Background information

When you use DCDN, a large amount of network log data is generated. The real-time log delivery feature allows you to collect the logs that are generated by edge nodes in real time and deliver the logs to Log Service. Then, you can consume the logs in Log Service to monitor and identify business issues in an efficient manner.



Assets

- Dedicated projects and dedicated Logstores

  After you enable the real-time log delivery feature of DCDN, Log Service automatically creates a project named dcdn-edge-rtlog-region-random ID and a Logstore named dcdn-edge-rtlog.

> **Note** To ensure that DCDN real-time logs can be delivered to Log Service, do not delete dedicated projects or Logstores.

- Dedicated dashboards

> **Note** We recommend that you do not make changes to the dedicated dashboards because this may affect the usability of the dashboards. You can create a custom dashboard to visualize log analysis results. For more information, see Create a dashboard.

Dashboard	Description
DCDN Access Center	Shows the real-time changes of DCDN business traffic. The information displayed on the dashboard includes the total number of page views (PVs), the total number of unique visitors (UVs), the percentage of error requests, top 10 URIs, top 10 IP addresses, the trend of error codes, and top 10 error domain names.

Billing

- You are charged when you use the real-time log delivery feature to collect logs. The charges are included into your DCDN bills. For more information, see Pricing of value-added service - Real-time log entries.
- After DCDN real-time logs are delivered to Log Service, you are charged based on the storage space, read traffic, the number of requests, data transformation, and data shipping. The charges are included into your Log Service bills. For more information, see Log Service pricing.

Limits

- You can create one or more delivery projects for a domain name. You cannot bind different delivery projects of a domain name to the same region, or to regions in and outside the Chinese mainland at the same time. Logs are collected from the regions to which the delivery project is bound. For example, if you create two delivery projects for the domain name example.com, you cannot bind these delivery projects to **US and Mainland China**, or to **Mainland China**. You can bind the delivery projects to **US and India**.
- Only one delivery project can be bound to a region that is supported by Log Service.

Benefits

- High timeliness: After you enable the real-time log delivery feature, log data is automatically delivered to Log Service. You can view the real-time log analysis results for logs that are generated within the previous 3 minutes in the Log Service console.
- Multiple fields: Real-time logs contain multiple log fields. For more information, see Log fields.
- Business compliance: The collection and delivery processes of real-time logs are compliant.
- Powerful data analysis: Log Service can work together with DCDN. Visualized analysis templates are generated for logs after the logs are delivered from DNCN to Log Service. This helps you analyze and monitor your business changes in an efficient manner. Scenarios:
  - Fine-grained analysis on URLs: You can analyze the trend in the number of requests from a URL, the profile of the requests from a specific location or source IP address, and the availability of a URL.

- Analysis on business reliability: You can analyze the error codes that are returned from requested URLs, error domain names, error URLs, and error IP addresses.
- Analysis on networks: You can analyze the response latency and cache hit ratio of networks.
- Profile analysis: You can analyze profiles based on geographic locations, terminals, and source IP addresses.

4.24.2. Enable the real-time log delivery feature

The real-time log delivery feature allows you to collect logs that are generated by Dynamic Route for CDN (DCDN) accelerated domain names in real time and deliver the collected logs to Log Service for analysis. This helps you monitor and identify business issues in an efficient manner. This topic describes how to enable the real-time log delivery feature.

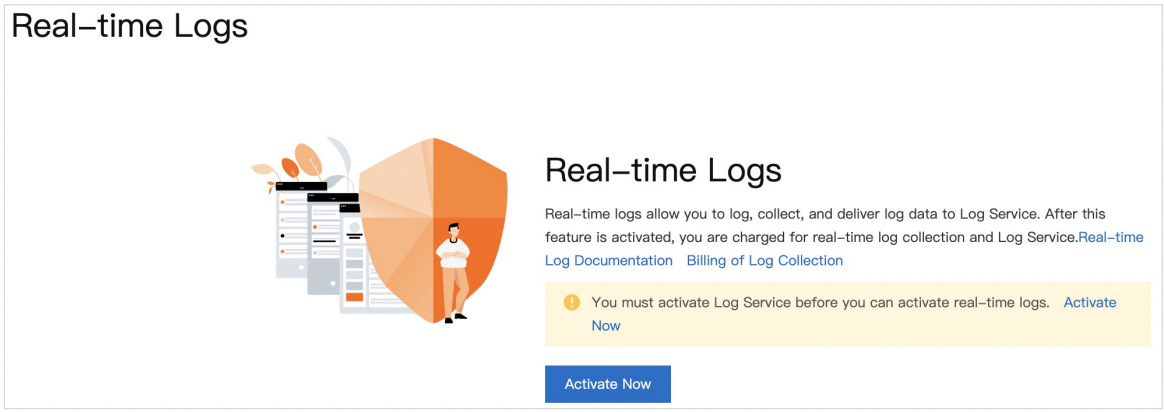
Prerequisites

- A domain name is added. For more information, see Add a domain name.
- Log Service is activated.

Procedure

1.
2. In the left-side navigation pane, choose **Data Center > Logs > Real-time Logs**.
3. Enable the real-time log delivery feature.

   The first time you use the real-time log delivery feature, enable the feature as prompted in the DCDN console.



4. On the **Real-time Logs** page, perform the following steps to enable the real-time log delivery feature.
   i. On the **Real-time Logs** page, click **Create Delivery Project**.

ii. In the **Create Real-time Log Delivery Project** dialog box, set the parameters and click **Next**. The following table describes the parameters.

Parameter	Description
Project Name	The project name is used to identify the delivery project and must be unique among other existing projects.
Log Field	Select log fields that you want to include in each real-time log. For more information, see Log fields.
Sampling Rate	Specify the percentage of the log entries that you want to deliver to Log Service. Valid values: 0 to 100%. **Note** The number of log entries to be delivered is approximately equal to the product of the number of full log entries and the specified sampling rate. If you want to deliver all log entries, set the Sampling Rate parameter to 100%.

iii. In the **Create Real-time Log Delivery Project** dialog box, specify regions for the Collected From and Delivered To parameters, and then click **Next**.

> **Notice** After you create a delivery project for real-time logs, you cannot change the values of the **Collected From** and **Delivered To** parameters. If you want to change the values, you must delete the delivery project that you created and create another project.

Parameter	Description
Collected From	The region from which logs are collected.
Delivered To	The region where the Log Service project resides. Log Service creates a project in the specified region to manage real-time logs.
Authorize	The first time you create a delivery project, you must authorize DCDN to assume the AliyunServiceRoleForDCDNRealTimeLogDelivery service-linked role to access Log Service resources. For more information, see SLR for real-time logs.

DCDN real-time logs are collected and then delivered to a project that resides in the specified region for storage. The following table describes the regions that are supported by the real-time log delivery feature.

Collected from	Delivered to
Mainland China	■ China (Hangzhou) ■ China (Shanghai) ■ China (Qingdao) ■ China (Beijing) ■ China (Zhangjiakou) ■ China (Shenzhen)
Europe	Germany (Frankfurt)
US	US (Silicon Valley)
India	India (Mumbai)
Other Country/Region	Singapore (Singapore)

iv. In the **Create Real-time Log Delivery Project** dialog box, select domain names and click **Next**.

What's next

After DCDN real-time logs are delivered to Log Service, you can query, analyze, download, ship, and transform the logs. You can also configure alert rules for the logs. For more information, see Common operations on logs of Alibaba Cloud services.

4.24.3. Log fields

This topic describes the fields of DCDN real-time logs.

Log field	Description
unixtime	The timestamp of a request.
domain	The request domain name.
method	The request method.
scheme	The protocol over which a request is sent.
uri	The resource that is requested.
uri_param	The request parameter.
client_ip	The real IP address of an end user. The value can be a public IP address or a LAN IP address.
proxy_ip	The IP address of a proxy.
remote_ip	The public IP address of the client that connects to a DCDN node.
remote_port	The port to which a DCDN node sends requests over the Internet.
refer_protocol	The protocol in the Referer HTTP header.
refer_domain	The domain name in the Referer HTTP header.
refer_uri	The URI in the Referer HTTP header.
refer_param	The parameters in the Referer HTTP header.
request_size	The size of a request that includes the request body and the request header.
request_time	The response time. Unit: milliseconds.
response_size	The size of a response. Unit: bytes.
return_code	The HTTP status code that is returned.
sent_http_content_range	The Content-Range header in an HTTP response. The value is specified by the origin server, for example, bytes=0-99/200.
server_addr	The IP address of the DCDN node that responds to a request.
server_port	The port on the DCDN node that responds to a request.
body_bytes_sent	The size of a request body. Unit: bytes.
content_type	The type of the requested resource.

Log field	Description
hit_info	The cache hit result. The cache hit results of requests for live streaming resources or dynamic content are not included. Valid values: • *HIT*: indicates a cache hit. • *MISS*: indicates a cache miss.
http_range	The value of the Range field in a request header. Example: bytes=0-100.
user_agent	The User-Agent HTTP header.
user_info	The user information.
uuid	The unique identifier of a request.
via_info	The Via HTTP header.
xforwordfor	The X-Forwarded-For (XFF) HTTP header.

5.Data import

5.1. Import data from OSS to Log Service

You can upload log files to Object Storage Service (OSS) for storage. Then, you can import the log data from OSS to Log Service and perform supported operations on the data in Log Service. For example, you can query, analyze, and transform the data. You can import only the OSS objects that are no more than 5 GB in size to Log Service. If you want to import a compressed object, the size of the compressed object must be no more than 5 GB.

Prerequisites

- Log files are uploaded to an OSS bucket. For more information, see Upload objects.
- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.
- Log Service is authorized to assume the AliyunLogImportOSSRole role to access your OSS resources. You can complete the authorization on the Cloud Resource Access Authorization page.

  If you use a RAM user, you must grant the PassRole permission to the RAM user. The following example shows a policy that you can use to grant the PassRole permission. For more information, see Create a custom policy and Grant permissions to a RAM user.

```
{
   "Statement": [
    {
      "Effect": "Allow",
      "Action": "ram:PassRole",
      "Resource": "acs:ram:*:*:role/aliyunlogimportossrole"
    }
  ],
  "Version": "1"
}
```

Create a data import configuration

> **Note** OSS objects of the Normal type support object-level marks. After an object of this type is modified, all data of the object is imported to Log Service. OSS objects of the Appendable type support row-level marks. After an object of this type is modified, only the appended data is imported to Log Service.

1.
2. On the **Data Import** tab in the **Import Data** section, click **OSS - Data Import**.
3. Select the project and Logstore. Then, click **Next**.
4. In the Configure Import Settings step, create a data import configuration.
   i. In the **Specify Data Source** step, configure the parameters. The following table describes the parameters.

Parameter	Description
Config Name	The name of the data import configuration.
OSS Region	The region where the OSS bucket resides. The OSS bucket stores the OSS objects that you want to import to Log Service. If the OSS bucket and the Log Service project reside in the same region, no Internet traffic is generated, and data is transferred at a high speed.
Bucket	The OSS bucket.
Folder Prefix	The directory of the OSS objects. If you configure this parameter, the system can find the OSS objects that you want to import in a more efficient manner. For example, if the OSS objects that you want to import are stored in the *csv/* directory, you can set this parameter to *csv/*. If you leave this parameter empty, the system traverses the entire OSS bucket to find the OSS objects. **Note** Cold Archive objects cannot be imported. If you want to import Cold Archive objects, you must restore the objects by using OSS API operations first. For more information, see Cold Archive.
Regular Expression Filter	The regular expression that is used to filter OSS objects. Only the objects whose names match the regular expression are imported. The names include the paths of the objects. By default, this parameter is empty, which indicates that no filtering is performed. For example, if an OSS object that you want to import is named *testdata/csv/bill.csv*, you can set this parameter to **(testdata/csv/)(.*)**.

Parameter	Description
Data Format	The format of the OSS objects. Valid values: ■ CSV: You can specify the first line of an OSS object as field names or specify custom field names. All lines except the first line are parsed as the values of log fields. ■ Single-line JSON: An OSS object is read line by line. Each line is parsed as a JSON object. The fields in JSON objects are log fields. ■ Parquet: An OSS object is automatically parsed into the format that is supported by Log Service. You do not need to configure further settings. If you use this format, you cannot preview data. ■ Single-line Text: Each line in an OSS object is parsed as a log. ■ Multi-line Text: Multiple lines in an OSS object are parsed as a log. You can specify a regular expression to match the first line or the last line for a log.
Compression Format	The compression format of the OSS objects that you want to import. Log Service decompresses the OSS objects based on the specified format to read data.
Encoding Format	The encoding format of the OSS objects that you want to import.
Restore Archived Files	If the OSS objects are Archive objects, Log Service cannot read data from the objects unless the objects are restored. If you turn on this switch, Archive objects are automatically restored.

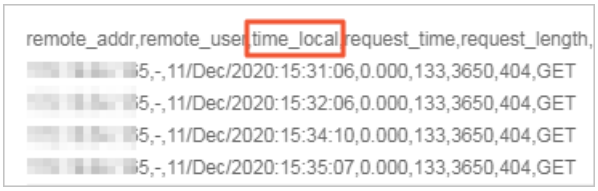
ii. Click **Preview**. The preview results are displayed.

iii. Confirm the preview results and click **Next**.

iv. In the **Specify Data Type** step, configure the parameters. The following tables describe the parameters.

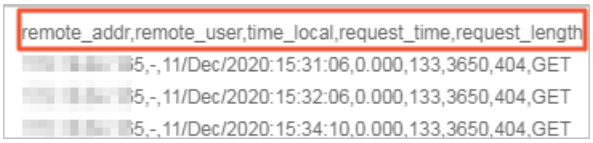
- Parameters related to log time

Parameter	Description

Parameter	Description
Use System Time	Specify whether to use the system time. ■ If you turn on **Use System Time**, the time field of a parsed log indicates the system time at which the log is imported. ■ If you turn off **Use System Time**, you must manually specify the time field and time format. **Note** We recommend that you turn on Use System Time. You can configure an index for the time field and use the index to query logs. If you import historical data to a Logstore and the data was generated earlier than the current time minus the data retention period of the Logstore, you cannot query the data in the Log Service console. For example, if you import data that was generated seven days ago to a Logstore whose data retention period is seven days, no results are returned when you query the data in the Log Service console.
Regex to Extract Time	If you select Single-line Text or Multi-line Text for Data Format and turn off **Use System Time**, you must specify a regular expression to extract log time. For example, if a sample log from a log file is **127.0.0.1 - - [10/Sep/2018:12:36:49 0800] "GET /index.html HTTP/1.1"**, you can set **Regex to Extract Time** to **[0-9]{0,2}\/[0-9a-zA-Z]+\/[0-9:,]+**.
Time Field	If you select CSV, Single-line JSON, or Parquet for Data Format and turn off **Use System Time**, you must specify a time field. For example, if the preview results of a CSV file display data as shown in the following figure, you can set **Time Field** to **time_local**. 

Parameter	Description
Time Format	If you turn off **Use System Time**, you must specify a time format that is supported by Java SimpleDateFormat. The time format is used to parse the time field. For more information about the time format syntax, see Class SimpleDateFormat. For more information about the common time formats, see Time formats. **Note** Java SimpleDateFormat does not support UNIX timestamps. If you want to use UNIX timestamps, you can set this parameter to epoch.
Time Zone	If you turn off **Use System Time**, you must specify a time zone. The time zone is used to parse log time to obtain time zone information. If the log time that is extracted includes time zone information, this parameter becomes invalid.

- Other parameters
  - Unique parameters when you set Data Format to CSV

Parameter	Description
Delimiter	The delimiter for logs. The default value is a comma (,).
Quote	The quote that is used to enclose a log field if the log field contains delimiters. The default value is double quotation marks (").
Escape Character	The escape character for logs. The default value is a backslash (\).
Max Lines for Multiline Logging	The maximum number of lines allowed for a log if the original log has multiple lines. Default value: 1.
First Line as Field Name	If you turn on **First Line as Field Name**, the first line in a CSV file is used to extract field names. For example, the first line in the CSV file that is shown in the following figure is used to extract field names. 
Custom Fields	If you turn off **First Line as Field Name**, you can specify custom field names based on your business requirements. Separate multiple field names with commas (,).
Lines to Skip	The number of lines that are skipped. For example, if you set this parameter to 1, the first line of a CSV file is skipped, and log collection starts from the second line.

- Unique parameters when you set Data Format to Multi-line Text

Parameter	Description
Position to Match with Regex	The usage of a regular expression. ■ If you select **Regex to Match First Line Only**, the regular expression that you specify is used to match the first line for a log. The unmatched lines are collected as a part of the log until the maximum number of lines that you specify is reached. ■ If you select **Regex to Match Last Line Only**, the regular expression that you specify is used to match the last line for a log. The unmatched lines are collected as a part of the next log until the maximum number of lines that you specify is reached.
Regular Expression	The regular expression. You can specify the regular expression based on log content. For more information, see How do I modify a regular expression?.
Max Lines	The maximum number of lines allowed for a log.

v. After you configure the parameters, click **Test**.

vi. After the test succeeds, click **Next**.

vii. In the **Specify Scheduling Interval** step, configure the parameters. The following table describes the parameters.

Parameter	Description
Import Interval	The interval at which the OSS objects are imported to Log Service.
Import Now	If you turn on **Import Now**, the OSS objects are immediately imported.

viii. Click **Next**.

5. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

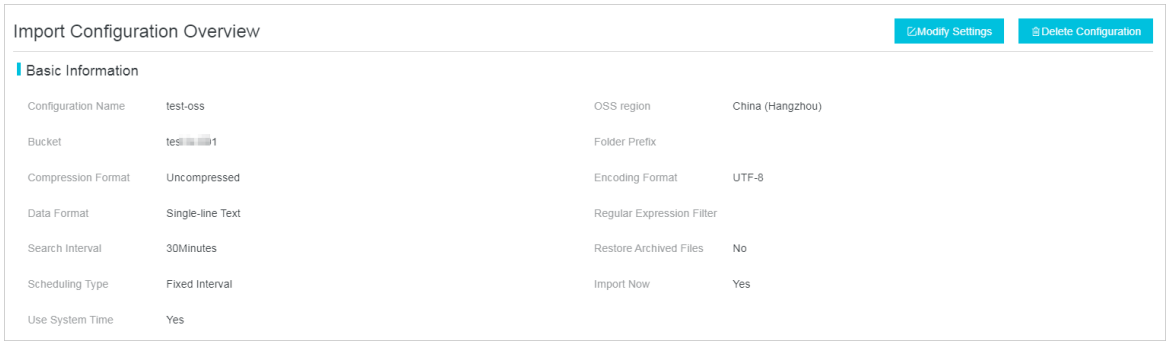
   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

6. Click **Next** to complete the creation.

View the data import configuration

After you create the data import configuration, you can view the configuration details and related statistical reports in the Log Service console.

1. In the **Projects** section, click the project to which the data import configuration belongs.
2. In the left-side navigation pane, choose **Log Storage > Logstores**. Click the Logstore to which the data import configuration belongs, choose **Data Import > Data Import**, and then click the name of the data import configuration.
3. On the **Import Configuration Overview** page, view the basic information and statistical reports of the data import configuration.



What to do next

On the **Import Configuration Overview** page, you can perform the following operations on the data import configuration:

- Modify the configuration

  Click **Modify Settings** to modify the data import configuration. For more information, see Create a data import configuration.

- Delete the configuration

  Click **Delete Configuration** to delete the data import configuration.

  > **Warning** After a data import configuration is deleted, it cannot be restored. Proceed with caution.

FAQ

Issue	Cause	Solution
I cannot select an OSS bucket when I create a data import configuration.	The AliyunLogImportOSSRole role is not assigned to Log Service.	Complete authorization based on the descriptions in the "Prerequisites" section of this topic.
Data cannot be imported.	The sizes of some OSS objects exceed 5 GB.	Reduce the sizes of the OSS objects.
After data is imported, I cannot query or analyze the data.	No indexes are configured, or configured indexes do not take effect.	Before you import data, we recommend that you configure indexes for the Logstore to which you want to import the data. For more information, see Configure indexes. After the issue occurs, you can reconfigure indexes for your Logstore. For more information, see Reindex logs for a Logstore.

Issue	Cause	Solution
Archive objects cannot be imported.	Restore Archived Files is turned off.	• Method 1: Modify the data import configuration and turn on **Restore Archived Files**. • Method 2: Create a data import configuration and turn on **Restore Archived Files**.
The **Regular Expression Filter** parameter is specified, but no data is collected.	• The specified regular expression is invalid. • A large number of OSS objects are stored in the OSS bucket. The system does not match OSS objects by traversing the OSS bucket before timeout.	Reconfigure the **Regular Expression Filter** parameter. If the issue persists, the cause may be that a large number of OSS objects are stored in the OSS bucket. In this case, specify a more specific directory of the OSS objects to reduce the number of objects that are involved in traversing.
Logs are imported, but no data is found in the Log Service console.	The log time goes beyond the data retention period of the Logstore. Expired data is deleted.	Check the time range for query and the data retention period of the Logstore.
The extracted log time is used to query data, but no data is found for that time.	The specified time format is invalid.	Check whether the time format is supported by Java SimpleDateFormat. For more information, see Class SimpleDateFormat.
An error occurred in parsing an OSS object that is in the Multi-line Text format.	The specified regular expression that is used to match the first line or the last line for a log is invalid.	Specify a valid regular expression.

Issue	Cause	Solution
The import speed suddenly slows down.	• The data that needs to be imported is not sufficient. • A large number of OSS objects are stored in the OSS bucket. A large amount of time is consumed to traverse the OSS bucket for the OSS objects.	1. Check whether the OSS bucket has sufficient data that needs to be imported. 2. Check whether a large amount of time is consumed to traverse the OSS bucket for the OSS objects because a large number of OSS objects are stored in the OSS bucket. If the issue occurs due to this reason, you can configure the **Folder Prefix** and **Regular Expression Filter** parameters to reduce the number of OSS objects that are involved in traversing. Alternatively, you can migrate the OSS objects that have been imported to Log Service from the OSS bucket to a different directory or bucket in the OSS console.

5.2. Import data from MaxCompute to Log Service

This topic describes how to import data from MaxCompute to Log Service. After you store log files in MaxCompute, you can import these files as MaxCompute data to Log Service. Then you can search, analyze, and transform the data in Log Service.

Prerequisites

- MaxCompute is activated and log files are uploaded to MaxCompute. For more information, see Import data to tables.
- A project and a Logstore are created. For more information, see Create a project and a Logstore.

Import data

1.
2. In the **Import Data** section, click the **MaxCompute - Data Import** card.
3.
4. Configure data import.

i. On the **Specify Data Source** tab, set the parameters. The following table describe the parameters.

Parameter	Description
Config Name	The name of the Logtail configuration file.
MaxCompute Project	The name of the MaxCompute project where the data to be imported resides.
Table	The name of the table where the data to be imported resides.
Partition Description	The description of MaxCompute table partitions. You must specify this parameter if you import data from a partitioned table. For more information, see Partition. Separate multiple-level partitions with commas (,).
AccessKey ID	The AccessKey ID that you use to access MaxCompute. **Note** The Alibaba Cloud account or RAM user to which the AccessKey ID belongs must be granted access to the MaxCompute project.
AccessKey Secret	The AccessKey secret that you use to access MaxCompute.
Endpoint	The endpoint of the MaxCompute project. For more information, see Endpoints.
Tunnel Endpoint	The tunnel endpoint that you use to access the MaxCompute project. For more information, see Endpoints.

ii. Click **Preview** to preview the data.

iii. Click **Next**.

iv. On the **Specify Data Type** tab, set the required parameters. The following table describes the parameters.

Parameter	Description

Parameter	Description
Use System Time	■ If you turn on the **Use System Time** switch, the time field of a parsed log entry is the system time when the log entry is imported. ■ If you turn off the **Use System Time** switch, you must manually configure the time field and format. **Note** We recommend that you turn on the **Use System Time** switch. You can configure an index for the time field and use the index for log queries. If you import data that is generated earlier than the current time minus the data retention period of the Logstore, the data cannot be queried in the Log Service console. For example, if the data retention period is seven days and you import data that was generated seven days ago, no results can be found in the Log Service console.
Time Field	If you turn off the **Use System Time** switch, you must specify a field to extract the log time.
Time Format	If you turn off the **Use System Time** switch, you must specify a time format by using the Java SimpleDateFormat class. The time format is used to parse time fields, or parse strings that are extracted by regular expressions. For more information, see Class SimpleDateFormat and Time formats. **Note** The Java SimpleDateFormat class does not support Unix timestamps. If you want to use Unix timestamps, you can set the time format to epoch.
Time Zone	If you turn off the **Use System Time** switch, you must specify a time zone to parse the time zone of the log time. This parameter is invalid if the time zone information already exists in the log format.

v. (Optional)After you specify the required parameters, click **Test**.

vi. After the test succeeds, click **Next**.

vii. (Optional)On the **Specify Scheduling Interval** tab, set the required parameters. The following table describes the parameters.

Parameter	Description
Import Interval	The interval at which the MaxCompute data is imported to Log Service.
Import Now	If you turn on the **Import Now** switch, the MaxCompute data is immediately imported.

viii. Click **Next**.

5. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

View the data import configurations

After you create data import configurations, you can view the configuration details and relevant statistical report in the Log Service console.

1. In the Projects section, click the target project.
2. Find the Logstore to which the data import configurations belong, choose **Data Import > Data Import**, and then click the name of the data import configurations.
3. On the **Import Configuration Overview** page, view the configuration details and statistical report.

References

On the **Import Configuration Overview** page, you can perform the following operations:

- Modify the data import configurations

  Click **Modify Settings** to modify the data import configurations. For more information, see Configure data import.

- Delete the data import configurations.

  You can click **Delete Configuration** to delete the data import configurations.

  > **Note** The data import configurations cannot be recovered after they are deleted.

5.3. Time formats

When you create an import task, you must specify a format for the time field. This topic describes the syntax of time formats and provides examples of time formats.

Time format syntax

Character	Description	Example
G	Era designator	AD
y	Year	2001
M	Month	July or 07
d	Day	10
h	Hour in AM or PM (1 to 12)	12
H	Hour in day (0 to 23)	22
m	Minute	30
s	Second	55
S	Millisecond	234
E	Week	Tuesday
D	Day in year	360
F	Day of week in month	2
w	Week in year	40
W	Week in month	1
a	AM or PM	PM
k	Hour in day (1 to 24)	24
k	Hour in AM or PM (0 to 11)	10
z	Time zone	Eastern Standard Time
'	Delimiter	Delimiter
"	Single quotation marks	"

Time format examples

Date format	Parsing syntax	Parsed value (unit: seconds)
2020-05-02 17:30:30	yyyy-MM-dd HH:mm:ss	1588411830
2020-05-02 17:30:30:123	yyyy-MM-dd HH:mm:ss:SSS	1588411830
2020-05-02 17:30	yyyy-MM-dd HH:mm	1588411800
2020-05-02 17	yyyy-MM-dd HH	1588410000

Date format	Parsing syntax	Parsed value (unit: seconds)
20-05-02 17:30:30	yy-MM-dd HH:mm:ss	1588411830
2020-05-02T17:30:30V	yyyy-MM-dd'T'HH:mm:ss'V'	1588411830
Sat May 02 17:30:30 CST 2020	EEE MMM dd HH:mm:ss zzz yyyy	1588411830

6.Other collection methods

6.1. Use web tracking to collect logs

Log Service allows you to collect logs from the HTML, HTML5, iOS, and Android platforms by using web tracking. Log Service also allows you to customize dimensions and metrics to collect logs. This topic describes how to collect logs by using web tracking.

Context

You can use web tracking to collect user information from browsers, iOS apps, or Android apps. The information includes:

- Browsers, operating systems, and resolutions that are used by users.
- User browsing behavior, such as the number of clicks and purchases on a website.
- The amount of time that users spend on an app and whether users are active users.

Usage notes

- After you enable web tracking for a Logstore, the write permissions on the Logstore are granted to anonymous users from the Internet. This may generate dirty data.
- The HTTP body of each GET request cannot exceed 16 KB.
- You can call the PutLogs API operation by using the POST method to write a maximum of 3 MB or 4,096 log entries to Log Service. For more information, see PutLogs.

Step 1: Enable web tracking

- Enable web tracking in the Log Service console.
  i. Log on to the Log Service console.
  ii. In the **Projects** section, click the project in which you want to enable web tracking for a Logstore.
  iii. On the **Log Storage > Logstores** tab, find the Logstore and choose > **Modify**.
  iv. On the **Logstore Attributes** page, click **Modify** in the upper-right corner of the page.
  v. Turn on the **WebTracking** switch and click **Save**.
- Enable web tracking by using an SDK.

  The following script shows how to enable web tracking by using Log Service SDK for Java.

```
import com.aliyun.openservices.log.Client;
import com.aliyun.openservices.log.common.LogStore;
import com.aliyun.openservices.log.exception.LogException;
public class WebTracking {
  static private String accessId = "your accesskey id";
  static private String accessKey = "your accesskey";
  static private String project = "your project";
  static private String host = "log service data address";
  static private String logStore = "your logstore";
  static private Client client = new Client(host, accessId, accessKey);
  public static void main(String[] args) {
      try {
          // Enable web tracking for an existing Logstore.
          LogStore logSt = client.GetLogStore(project, logStore).GetLogStore();
          client.UpdateLogStore(project, new LogStore(logStore, logSt.GetTtl(), logSt.Get
ShardCount(), true));
          // Disable web tracking.
          //client.UpdateLogStore(project, new LogStore(logStore, logSt.GetTtl(), logSt.G
etShardCount(), false));
          // Create a Logstore for which you want to enable web tracking.
          //client.UpdateLogStore(project, new LogStore(logStore, 1, 1, true));
      }
      catch (LogException e){
          e.printStackTrace();
      }
  }
}
```

Step 2: Collect log data

After you enable web tracking for a Logstore, you can upload logs to a Logstore by using the following methods:

- Use SDK for JavaScript to upload logs.
  i. Install the dependency.

  ```
  npm install --save js-sls-logger
  ```

  ii. Import the application module.

  ```
  import SlsWebLogger from 'js-sls-logger'
  ```

  iii. Set the opts parameter.

  ```
  const opts = {
    host: 'cn-hangzhou.log.aliyuncs.com',
    project: 'my_project_name',
    logstore: 'my_logstore_name',
    time: 10,
    count: 10,
  }
  ```

Parameter	Required	Description
host	Yes	The endpoint of Log Service in the region where your Log Service project resides. For more information, see Endpoints. The China (Hangzhou) region is used as an example. For other regions, configure the endpoint based on the project name and the region where the project resides.
project	Yes	The name of the project.
logstore	Yes	The name of the Logstore.
time	No	The time interval at which messages are sent. Default value: 10. Unit: seconds.
count	No	The number of sent messages. Default value: 10.

iv. Create SlsWebLogger.

```
const logger = new SlsWebLogger(opts)
```

v. Upload logs.

```
logger.send({
    customer: 'zhangsan',
    product: 'iphone 12',
    price: 7998
})
```

- Use the GET method to upload logs.

  Run the following command to upload logs. Replace the parameters as needed.

```
curl --request GET 'http://${project}.${host}/logstores/${logstore}/track? APIVersion=0.6.0&key1=val1&key2=val2'
```

Parameter	Required	Description
${project}	Yes	The name of the project.
${host}	Yes	The endpoint of Log Service in the region where your project resides. For more information, see Endpoints.
${logstore}	Yes	The name of the Logstore.
APIVersion=0.6.0	Yes	A reserved parameter.
__topic__=yourtopic	No	The log topic.

Parameter	Required	Description
key1=val1&key2=val2	Yes	The key-value pairs that you want to upload to Log Service. Make sure that the data size is less than 16 KB.

- Use HTML <img> tags to upload logs.

```
<img src='http://${project}.${host}/logstores/${logstore}/track.gif?APIVersion=0.6.0&key1=val1&key2=val2'/>
<img src='http://${project}.${host}/logstores/${logstore}/track_ua.gif?APIVersion=0.6.0&key1=val1&key2=val2'/>
```

  The track_ua.gif file contains custom parameters that you want to upload to Log Service. If you use this method to upload logs, Log Service records both the custom parameters and the UserAgent and referer HTTPS headers as log fields.

  > **Note** To collect the referer header, make sure that the URL in the second <img> tag uses the HTTPS protocol.

- Use the POST method to upload logs.

  You can send an HTTP POST request to upload a large amount of data. For more information, see PutWebtracking.

6.2. Use the Kafka protocol to upload logs

You can use Kafka Producer SDKs or collection agents to collect logs, and then upload the logs to Log Service by using the Kafka protocol. This topic describes how to upload logs to Log Service by using the Kafka protocol.

Limits

- Only Kafka 0.8.0 to Kafka 2.1.1 (message format version 2) are supported.
- You must use the SASL_SSL protocol to ensure the security of log transfer.
- If your Logstore contains multiple shards, you must upload log data in load balancing mode.
- You can use only the Kafka protocol to upload logs that are collected by using Kafka Producer SDKs or collection agents.

Configurations

You must specify relevant parameters when you use the Kafka protocol to upload logs to Log Service. The following table describes the required parameters.

Parameter	Description
Connection protocol	Set the value to SASL_SSL.

Parameter	Description
hosts	The endpoint of the Log Service project. The format is *project name*.endpoint. Specify an endpoint based on the region where your Log Service project resides. For more information, see Endpoints. • Alibaba Cloud internal network: The port number is 10011. Example: test-project-1.cn-hangzhou-intranet.log.aliyuncs.com:10011. • Internet: The port number is 10012. Example: test-project-1.cn-hangzhou.log.aliyuncs.com:10012.
topic	The name of the Logstore.
username	The name of the project.
password	The AccessKey pair. The format is ${access-key-id}#${access-key-secret}. Enter your AccessKey ID in ${access-key-id} and your AccessKey secret in ${access-key-secret}. We recommend that you use the AccessKey pair of your RAM user. For more information, see Authorize a RAM user to connect to Log Service.
Certificate file	The domain name of each Log Service project has a CA certificate. You only need to download the root certificate and save the certificate to a directory, for example, */etc/ssl/certs/ca-bundle.crt*.

Example 1: Use Beats to upload logs to Log Service

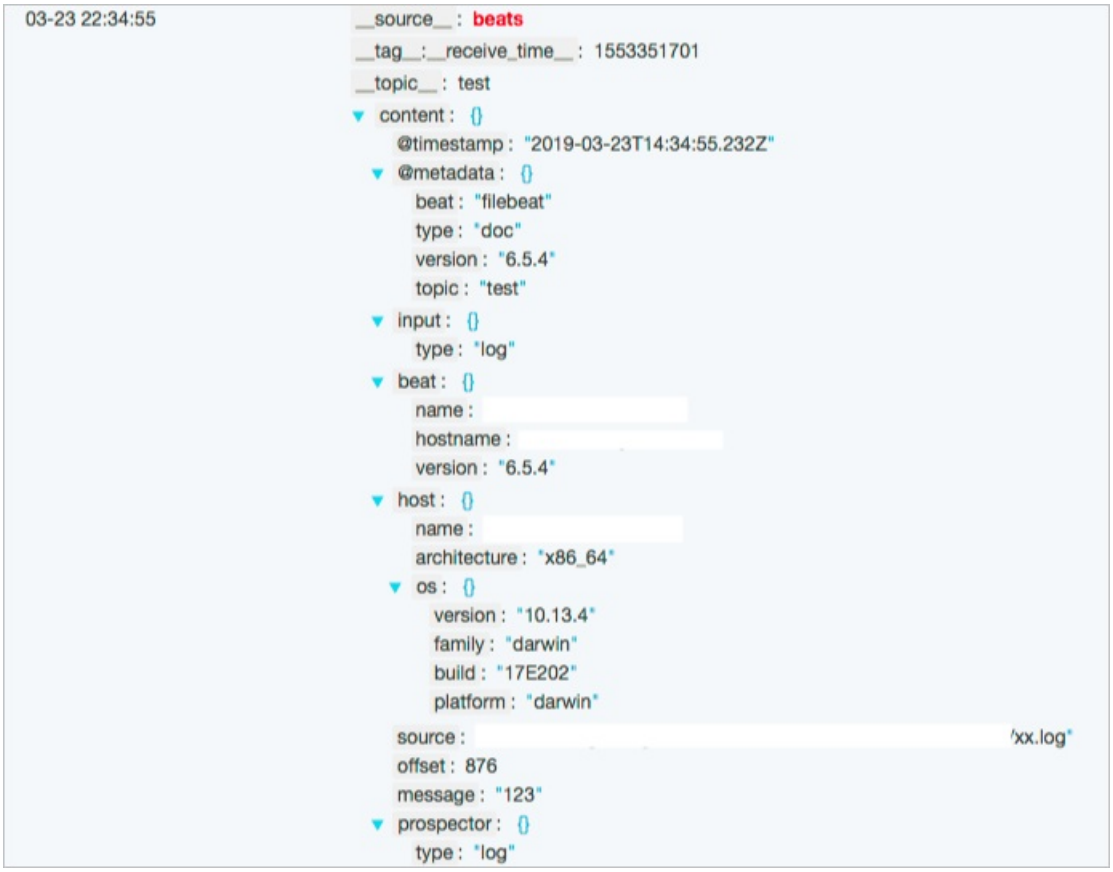
You can use Beats such as MetricBeat, PacketBeat, Winlogbeat, Auditbeat, Filebeat, and Heartbeat to collect logs. After the logs are collected, you can use the Kafka protocol to upload the logs to Log Service. For more information, visit Beats-Kafka-Output.

- Configuration example

```
output.kafka:
  # initial brokers for reading cluster metadata
  hosts: ["test-project-1.cn-hangzhou.log.aliyuncs.com:10012"]
  username: "yourusername"
  password: "yourpassword"
  ssl.certificate_authorities:
  # message topic selection + partitioning
  topic: 'test-logstore-1'
  partition.round_robin:
    reachable_only: false
  required_acks: 1
  compression: gzip
  max_message_bytes: 1000000
```

- Sample log entry

  By default, Beats provide JSON-formatted logs in the content field. You can create a JSON index for the content field. For more information, visit JSON type.



Example 2: Use collectd to upload logs to Log Service

collectd is a daemon that is used to collect the performance metrics of systems and applications at a regular interval. The collectd daemon allows you to upload logs to Log Service by using the Kafka protocol. For more information, visit Write Kafka Plug-in.

Before you upload logs to Log Service, you must install the collectd-write_kafka plug-in and relevant dependencies. If you are using CentOS, you can run the `sudo yum install collectd-write_kafka` command to install the collectd-write_kafka plug-in. For more information, visit RPM resource collectd-write_kafka.

- Configuration example

  In the following configuration example, the output format of logs is set to JSON. You can also set the output format to Command and Graphite. For more information, visit Manpage collectd.conf.

```
<Plugin write_kafka>
  Property "metadata.broker.list" "test-project-1.cn-hangzhou.log.aliyuncs.com:10012"
  Property "security.protocol" "sasl_ssl"
  Property "sasl.mechanism" "PLAIN"
  Property "sasl.username" "yourusername"
  Property "sasl.password" "yourpassword"
  Property "broker.address.family" "v4"
  <Topic "test-logstore-1">
    Format JSON
    Key "content"
  </Topic>
</Plugin>
```

- Sample log entry

  Logs are sent to the Log Service console in the JSON format. The log content is included in the content field. You can create a JSON index for the content field. For more information, see JSON type.

Example 3: Use Telegraf to upload logs to Log Service

Telegraf is an agent in the Go programming language and is used to collect, process, and analyze metrics. It consumes only a small number of memory resources. In addition, Telegraf supports integration with multiple plug-ins. You can use Telegraf to retrieve metrics from the system where it runs, or from third-party APIs. You can also use Telegraf to monitor metrics by using StatsD and Kafka consumers.

Before you use the Kafka protocol to upload collected logs to Log Service, you must modify the configuration file of Telegraf.

- Configuration example

  In the following configuration example, the output format of logs is set to JSON. You can also set the output format to Graphite and Carbon2. For more information, visit Telegraf output formats.

  > **Note** You must set a valid *tls_ca* path for Telegraf. You can use the default directory of the root certificate of the server. In a Linux-based server, the default directory of the root certificate is */etc/ssl/certs/ca-bundle.crt*.

```
[[outputs.kafka]]
  ## URLs of kafka brokers
  brokers = ["test-project-1.cn-hangzhou.log.aliyuncs.com:10012"]
  ## Kafka topic for producer messages
  topic = "test-logstore-1"
  routing_key = "content"
  ## CompressionCodec represents the various compression codecs recognized by
  ## Kafka in messages.
  ## 0 : No compression
  ## 1 : Gzip compression
  ## 2 : Snappy compression
  ## 3 : LZ4 compression
  compression_codec = 1
  ## Optional TLS Config tls_ca = "/etc/ssl/certs/ca-bundle.crt"
  # tls_cert = "/etc/telegraf/cert.pem" # tls_key = "/etc/telegraf/key.pem"
  ## Use TLS but skip chain & host verification
  # insecure_skip_verify = false
  ## Optional SASL Config
  sasl_username = "yourusername"
  sasl_password = "yourpassword"
  ## Data format to output.
  ## https://github.com/influxdata/telegraf/blob/master/docs/DATA_FORMATS_OUTPUT.md
  data_format = "json"
```

- Sample log entry

  Logs are sent to the Log Service console in the JSON format. The log content is included in the content field. You can create a JSON index for the content field. For more information, see JSON type.

Example 4: Use Fluentd to upload logs to Log Service

Fluentd is an open source log data collector. It is a Cloud Native Computing Foundation (CNCF) project and complies with the Apache 2 License protocol.

Fluentd is compatible with multiple input plug-ins, processing plug-ins, and output plug-ins. You can use the fluent-plugin-kafka plug-in to upload logs to Log Service. For more information about how to install and configure this plug-in, visit fluent-plugin-kafka.

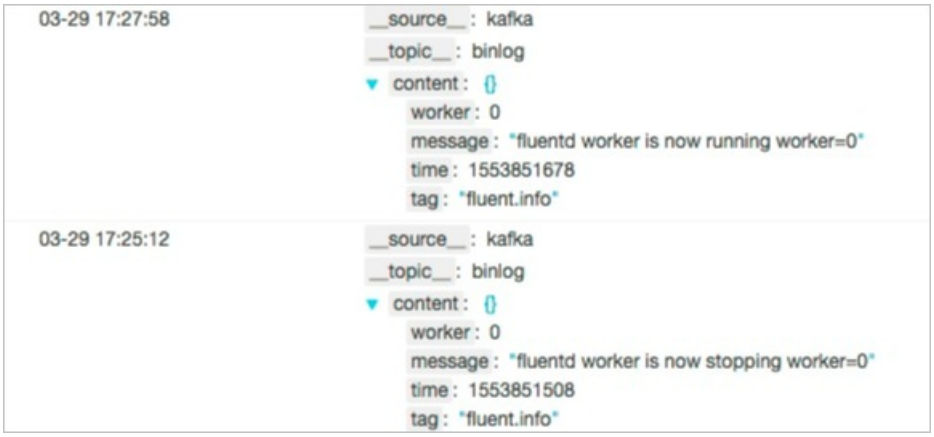
- Configuration example

  In the following configuration example, the output format of logs is set to JSON. Fluentd also supports other output formats. For more information, visit Fluentd Formatter.

```
<match **>
  @type kafka
  # Brokers: you can choose either brokers or zookeeper.
  brokers      test-project-1.cn-hangzhou.log.aliyuncs.com:10012
  default_topic test-logstore-1
  default_message_key content
  output_data_type json
  output_include_tag true
  output_include_time true
  sasl_over_ssl true
  username yourusername
  password yourpassword
  ssl_ca_certs_from_system true
  # ruby-kafka producer options
  max_send_retries 10000
  required_acks 1
  compression_codec gzip
</match>
```

- Sample log entry

  Logs are sent to the Log Service console in the JSON format. The log content is included in the content field. You can create a JSON index for the content field. For more information, see JSON type.



Example 5: Use Logstash to upload logs to Log Service

Logstash is an open source log collection engine that provides real-time processing capabilities. You can use Logstash to dynamically collect logs from different sources.

Logstash provides a built-in Kafka output plug-in. You can configure Logstash to upload logs to Log Service by using the Kafka protocol. Log Service uses the SASL_SSL protocol during data transfer. You must configure the SSL certificate and Java Authentication and Authorization Service (JAAS) file.

- Configuration example
  i. Create a JAAS file and save the file to a directory, for example, */etc/kafka/kafka_client_jaas.conf*.

     Add the following content to the JAAS file:

     ```
     KafkaClient {
       org.apache.kafka.common.security.plain.PlainLoginModule required
       username="yourusername"
       password="yourpassword";
     };
     ```

  ii. Configure the SSL certificate and save the certificate to a directory, for example, */etc/kafka/client-root.truststore.jks*.

     The domain name of each Log Service project has a CA certificate. You only need to download the root certificate GlobalSign Root CA, encode the certificate in Base64, and save the certificate to a directory, for example, */etc/kafka/ca-root*. Then, run the following keytool command to generate a JKS file. You must set a password when a JKS file is generated for the first time.

     ```
     keytool -keystore client.truststore.jks -alias root -import -file /etc/kafka/ca-root
     ```

  iii. Configure Logstash.

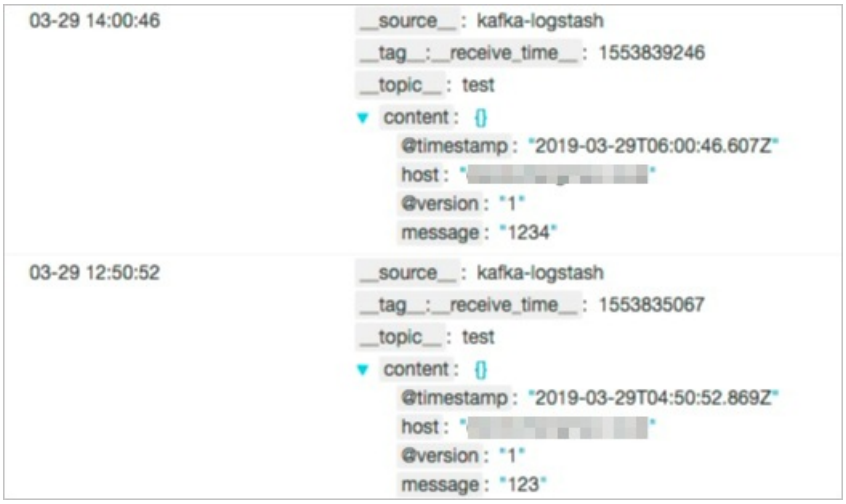
     In the following configuration example, the output format of logs is set to JSON. Logstash also supports other output formats. For more information, visit Logstash Codec.

     > **Note** The following configurations are used for a connectivity test. In a production environment, we recommend that you remove the stdout field.

     ```
     input { stdin { } }
     output {
       stdout { codec => rubydebug }
       kafka {
         topic_id => "test-logstore-1"
         bootstrap_servers => "test-project-1.cn-hangzhou.log.aliyuncs.com:10012"
         security_protocol => "SASL_SSL"
         ssl_truststore_location => "/etc/client-root.truststore.jks"
         ssl_truststore_password => "123456"
         jaas_path => "/etc/kafka_client_jaas.conf"
         sasl_mechanism => "PLAIN"
         codec => "json"
         client_id => "kafka-logstash"
       }
     }
     ```

- Sample log entry

Logs are sent to the Log Service console in the JSON format. The log content is included in the content field. You can create a JSON index for the content field. For more information, see JSON type.



Error messages

If a log fails to be uploaded by using the Kafka protocol, an error message is returned. The following table describes the error messages. For more information, see Exception summary.

Error message	Description	Solution
NetworkException	The error message is returned because a network exception has occurred.	Try again after 1 second.
TopicAuthorizationException	The error message is returned because the authentication has failed.	This is because your AccessKey pair is invalid or you are not authorized to write data to the destination project or Logstore. Enter a valid AccessKey pair and make sure that it has the required write permissions.
UnknownTopicOrPartitionException	The error message is returned because one of the following errors has occurred: • The destination project or Logstore does not exist. • The region of the project is different from the region of the specified endpoint.	Create a project and a Logstore in advance. Make sure that the region of the project is the same as the region of the specified endpoint.
KafkaStorageException	The error message is returned because a server error has occurred.	Try again after 1 second.

6.3. Use the syslog protocol to upload logs

You can use the rsyslog and syslog-ng utilities to collect logs. After the logs are collected, you can use the syslog protocol to upload the logs to Log Service. This topic describes how to upload logs to Log Service by using the syslog protocol.

Limits

- Syslog logs must be stored based on the RFC 5424 protocol.
- The maximum size of each log is 64 KB.
- Transport Layer Security (TLS) 1.2 must be used to ensure the security of data transmission.

Configurations

> **Notice** In most cases, you cannot use the TLS protocol or RFC 5424 protocol to collect log data from devices such as on-premises VPNs and routers. We recommend that you use the syslog plug-in of Logtail to collect data from the devices. For more information, see Collect syslogs.

If you upload logs by using the syslog protocol, you must specify the endpoint to which you want to upload the logs. The address is in the `<Project name>.<Log Service endpoint>:<Syslog protocol port number>` format. Example: **test-project-1.cn-hangzhou-intranet.log.aliyuncs.com:10009**. Specify an endpoint based on the region where your Log Service project resides. For more information, see Endpoints. The syslog port is 10009. You must also specify a Log Service project, a Logstore, and an AccessKey pair in the STRUCTURED-DATA field. The following table describes the parameters.

Parameter	Description	Example
STRUCTURED-DATA	The value is set to Logservice.	Logservice
Project	The name of a project. Before you can collect logs, you must create a project.	test-project-1
Logstore	The name of a Logstore. Before you can collect logs, you must create a Logstore.	test-logstore-1
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. For more information, see Create a RAM user and authorize the RAM user to access Log Service.	<yourAccessKeyId>

Parameter	Description	Example
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. For more information, see Create a RAM user and authorize the RAM user to access Log Service.	<yourAccessKeySecret>

Example 1: Use the rsyslog utility to upload syslog logs to Log Service

The rsyslog utility is pre-installed on Linux servers. You can use the rsyslog utility to collect system logs. Then, you can use the syslog protocol to upload the logs to Log Service. Different versions of rsyslog use different configuration files. You can run the **man rsyslogd** command to view the rsyslog version.

> **Note** The rsyslog utility must contain the gnutls module. If the utility does not include the module, you can run the **sudo apt-get install rsyslog-gnutls** command or **sudo yum install rsyslog-gnutls** command to install the module.

1. Open the rsyslog configuration file.

   The default path of the rsyslog configuration file is */etc/rsyslog.conf*.

2. Configure the following settings based on the version of your rsyslog and append the configurations to your rsyslog configuration file:

   - Rsyslog v8 or later

     Set the `$DefaultNetstreamDriverCAFile` parameter to the path of the root certificate in the system.

```
# Setup disk assisted queues
$WorkDirectory /var/spool/rsyslog # where to place spool files
$ActionQueueFileName fwdRule1     # unique name prefix for spool files
$ActionQueueMaxDiskSpace 1g       # 1gb space limit (use as much as possible)
$ActionQueueSaveOnShutdown on     # save messages to disk on shutdown
$ActionQueueType LinkedList       # run asynchronously
$ActionResumeRetryCount -1        # infinite retries if host is down
$ActionSendTCPRebindInterval 100  # close and re-open the connection to the remote ho
st every 100 of messages sent.
#RsyslogGnuTLS set to default ca path
$DefaultNetstreamDriverCAFile /etc/ssl/certs/ca-bundle.crt
template(name="LogServiceFormat" type="string"
string="<%pri%>1 %timestamp:::date-rfc3339% %HOSTNAME% %app-name% %procid% %msgid% [l
ogservice project=\"test-project-1\" logstore=\"test-logstore-1\" access-key-id=\"<yo
urAccessKeyId>\" access-key-secret=\"<yourAccessKeySecret>\"] %msg%\n"
)
# Send messages to Loggly over TCP using the template.
action(type="omfwd" protocol="tcp" target="test-project-1.cn-hangzhou.log.aliyuncs.co
m" port="10009" template="LogServiceFormat" StreamDriver="gtls" StreamDriverMode="1"
StreamDriverAuthMode="x509/name" StreamDriverPermittedPeers="*.cn-hangzhou.log.aliyun
cs.com")
```

- Rsyslog v7 or earlier

  Set the `$DefaultNetstreamDriverCAFile` parameter to the path of the root certificate in the system.

```
# Setup disk assisted queues
$WorkDirectory /var/spool/rsyslog       # where to place spool files
$ActionQueueFileName fwdRule1           # unique name prefix for spool files $ActionQ
ueueMaxDiskSpace 1g             # 1gb space limit (use as much as possible) $ActionQu
eueSaveOnShutdown on           # save messages to disk on shutdown
$ActionQueueType LinkedList             # run asynchronously
$ActionResumeRetryCount -1              # infinite retries if host is down $ActionSen
dTCPRebindInterval 100        # close and re-open the connection to the remote host e
very 100 of messages sent.
# RsyslogGnuTLS set to default ca path
$DefaultNetstreamDriverCAFile /etc/ssl/certs/ca-bundle.crt
$ActionSendStreamDriver gtls
$ActionSendStreamDriverMode 1
$ActionSendStreamDriverAuthMode x509/name
$ActionSendStreamDriverPermittedPeer test-project-1.cn-hangzhou.log.aliyuncs.com
template(name="LogServiceFormat" type="string" string="<%pri%>1 %timestamp:::date-rfc
3339% %HOSTNAME% %app-name% %procid% %msgid% [logservice project=\"test-project-1\" l
ogstore=\"test-logstore-1\" access-key-id=\"<yourAccessKeyId>\" access-key-secret=\"<
yourAccessKeySecret>\"] %msg%\n")
*.* action(type="omfwd" protocol="tcp" target="test-project-1.cn-hangzhou.log.aliyunc
s.com" port="10009" template="LogServiceFormat")
```

3. Restart the rsyslog utility.

   Run the **sudo service rsyslog restart** command, **sudo /etc/init.d/syslog-ng restart** command, or **systemctl restart rsyslog** command to restart the rsyslog utility.

4. Run the **logger** command to generate test logs.

   For example, you can run the **logger hello world!** command to generate logs.

Example 2: Use the syslog-ng utility to upload syslog logs to Log Service

Syslog-ng is an open source utility that runs on UNIX and UNIX-like systems. This utility is based on the syslog protocol. You can run the **sudo yum install syslog-ng** command or **sudo apt-get install syslog-ng** command to install the syslog-ng utility.

> **Note** The rsyslog utility is pre-installed on Linux servers. This utility is incompatible with the syslog-ng utility. Before you use the syslog-ng utility, you must uninstall the rsyslog utility.

1. Open the syslog-ng configuration file.

   The default path of the syslog-ng configuration file is */etc/syslog-ng/syslog-ng.conf*.

2. Configure the following settings and append the configurations to your syslog-ng configuration file:

```
### Syslog-ng Logging Config for LogService ###
template LogServiceFormat {
    template("<${PRI}>1 ${ISODATE} ${HOST:--} ${PROGRAM:--} ${PID:--} ${MSGID:--} [logservice project=\"test-project-1\" logstore=\"test-logstore-1\" access-key-id=\"<yourAccessKeyId>\" access-key-secret=\"<yourAccessKeySecret>\"] $MSG\n"); template_escape(no);

};
destination d_logservice{
     tcp("test-project-1.cn-hangzhou.log.aliyuncs.com" port(10009)
     tls(peer-verify(required-untrusted))
     template(LogServiceFormat));
};
log {
     source(s_sys); # default use s_sys
     destination(d_logservice);
};
### END Syslog-ng Logging Config for LogService ###
```

3. Restart the syslog-ng utility.

   Run the **sudo /etc/init.d/syslog-ng restart** command, **sudo service syslog-ng restart** command, or **sudo systemctl restart syslog-ng** command to restart the syslog-ng utility.

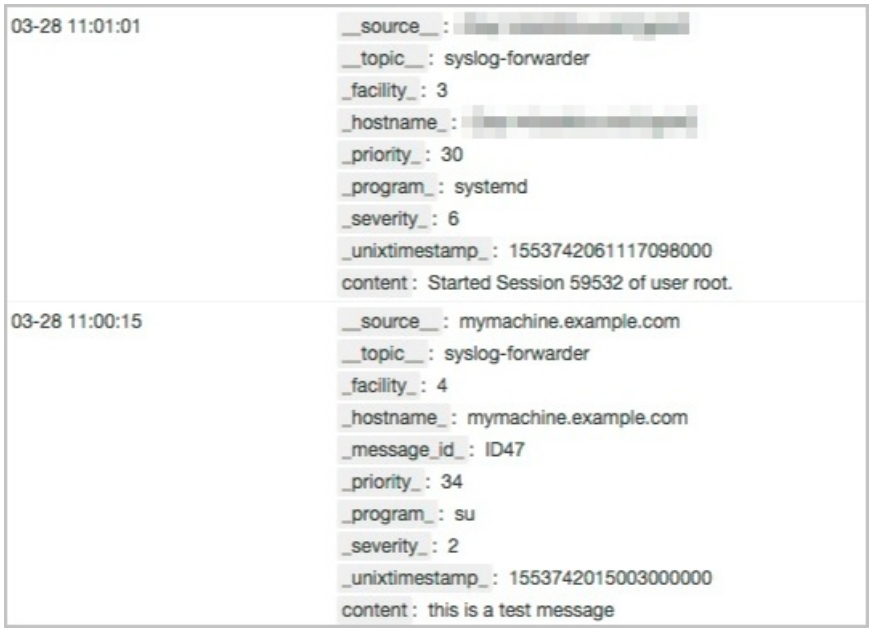
4. Run the **logger** command to generate test logs.

   For example, you can run the **logger hello world!** command to generate logs.

Sample logs

After you upload logs to Log Service, you can view the logs in the Log Service console. For information about log fields, see RFC 5424 protocol.

> **Note** By default, Log Service deletes the STRUCTURED-DATA field to ensure that your AccessKey pair is not leaked.



Log field	Description
__source__	The hostname in the raw log.
__topic__	The value is set to syslog-forwarder.
__facility__	The facility information, such as the information of the device and module.
__program__	The name of the process.
__serverity__	The severity level of the syslog log.
__priority__	The priority of the syslog log.
__unixtimestamp__	The UNIX timestamp of the raw log. Unit: nanoseconds.
content	The msg field in the raw log.

FAQ

- How do I simulate log uploading?

  You can use Netcat to simulate log uploading. This way, you can check whether the network connection is normal and whether the AccessKey pair is authorized to send syslog logs.

  i. Log on to the server on which you want to simulate log uploading.

  ii. Run the following command to install Netcat:

  ```
  sudo yum install nmap-ncat
  ```

  iii. Run the following command to connect to Log Service:

  ```
  ncat --ssl <yourProject>.<yourEndpoint> 10009
  ```

  Example:

  ```
  ncat --ssl test-project-1.cn-hangzhou.log.aliyuncs.com 10009
  ```

  iv. Netcat does not check whether network connections are interrupted. After you run a **ncat** command, enter the information that you want to send and press the Enter key in 30 seconds.

  ```
  <34>1 2019-03-28T03:00:15.003Z mymachine.example.com su - ID47 [logservice project="<yourProject>" logstore="<yourLogstore>" access-key-id="<yourAccessKeyID>" access-key-secret="<yourAccessKeySecret>"] this is a test message
  ```

  Example:

  ```
  <34>1 2019-03-28T03:00:15.003Z mymachine.example.com su - ID47 [logservice project="trace-doc-test" logstore="doc-test-001-logs" access-key-id="LTAI4***" access-key-secret="HfJEw***"] this is a test message
  ```

  v. After you send the syslog log, you can preview the log in the Log Service console.

  For more information, see Preview logs.

Time/Source	Content
2019-03-28 11:00:15 mymachine.example.com	content:this is a test message _hostname_:mymachine.example.com _severity_:2 _facility_:4 _message_id_:ID47 _unixtimestamp_:1553742015003000000 _program_:su _priority_:34

- What do I do if logs fail to be uploaded?

  Troubleshoot the failure based on the error message. For more information, see How do I view Logtail collection errors?.

- How do I view rsyslog error logs?

  You can run the **vim** command to view rsyslog error logs. By default, rsyslog error logs are stored in the */var/log/message* directory.

  - Error message 1

    ```
    dlopen: /usr/lib64/rsyslog/lmnsd_gtls.so: cannot open shared object file: No such file or directory
    ```

    This error message is returned because the rsyslog-gnutls module is not installed. You can run the **sudo apt-get install rsyslog-gnutls** command or **sudo yum install rsyslog-gnutls** command to install the module. After you install the module, restart the rsyslog utility.

  - Error message 2

    ```
    unexpected GnuTLS error -53 - this could be caused by a broken connection. GnuTLS reports:Error in the push function
    ```

    This error message is returned because the TCP connection is terminated because no actions are performed for a long period of time. You can ignore this error because rsyslog re-establishes the connection.

- How do I view syslog-ng error logs?

  You can run the **systemctl status syslog-ng.service** command or **journalctl -xe** command to view syslog-ng error logs. By default, syslog-ng error logs are stored in journal logs.

  If the following error message is returned, check whether the format of the configuration file is valid or whether configuration conflicts exist. For example, you cannot configure multiple `internal()` sources.

  ```
  Job for syslog-ng.service failed because the control process exited with error code. See "systemctl status syslog-ng.service" and "journalctl -xe" for details
  ```

6.4. Logstash

6.4.1. Install Logstash

This topic describes how to install Logstash.

Context

Logstash is an open source software application for data collection. You can use Logstash to collect logs and then use the logstash-output-logservice plug-in to upload the logs to Log Service. To

download the logstash-output-logservice plug-in, visit GitHub.

Procedure

1. Install the JDK package.
   i. Download the JDK package.

      Go to the Java official website, download the JDK package, and then double-click the package to install the JDK package.

   ii. Set the environment variables.

      Choose **Control Panel > System > Advanced system settings > Advanced > Environment Variables**. In the window that appears, set the environment variables.

      - PATH: *C:\Program Files\Java\jdk1.8.0_73\bin*
      - CLASSPATH: *C:\Program Files\Java\jdk1.8.0_73\lib;C:\Program Files\Java\jdk1.8.0_73\lib\tools.jar*
      - JAVA_HOME: *C:\Program Files\Java\jdk1.8.0_73*

      Replace *jdk1.8.0_73* with your actual JDK version.

   iii. Verify that the JDK package is installed.

      Run the **java -version** command. If a result that is similar to the following example is returned, the JDK package is installed.

      ```
      PS C:\Users\Administrator> java -version
      java version "1.8.0_73"
      Java(TM) SE Runtime Environment (build 1.8.0_73-b02)
      Java HotSpot(TM) 64-Bit Server VM (build 25.73-b02, mixed mode)
      PS C:\Users\Administrator> javac -version
      javac 1.8.0_73
      ```

2. Install Logstash.
   i. Download the Logstash installation package.

      > **Note** We recommend that you download Logstash 5.0 or later.

   ii. Decompress the downloaded package to the specified directory.
3. Install Logstash plug-in.

i. Select the installation mode based on the network environment of the server.

- Online installation

  The plug-in is hosted in the RubyGems service. For more information, see logstash-output-logservice.

  Run the following command to install Logstash:

  ```
  PS C:\logstash-6.4.3> .\bin\logstash-plugin install logstash-output-logservice
  ```

- Offline installation

  a. Download the installation package.

  b. Run the following command to install Logstash:

  ```
  bin/logstash-plugin install file:///root/logstash-offline-plugins.zip
  ```

ii. Verify that the Logstash plug-in is installed.

Run the following command. If the logstash-output-logservice plug-in is displayed in the returned plug-in list, the plug-in is installed.

```
PS C:\logstash-6.4.3> .\bin\logstash-plugin list
```

6.4.2. Create Logstash configurations for log collection and processing

This topic describes how to create Logstash configurations for log collection and processing.

Plug-ins

- logstash-input-file plug-in

  The logstash-input-file plug-in collects logs by using the tail command. For more information, visit logstash-input-file.

- logstash-output-logservice plug-in

  The logstash-output-logservice plug-in processes the collected logs and uploads the logs to Log Service.

Procedure

1. Create a configuration file in the *C:\logstash-2.2.2-win\conf* directory.

   Replace logstash-2.2.2-win with your actual Logstash version. You can create a configuration file for each type of log. The file name is in the **.conf* format.

2. Create configurations for log collection and processing.

   Create the following configurations for log collection and processing based on your business requirements and add the configurations to the configuration file. The configuration for log collection is specified by the input parameter. For more information, see Logstash documentation. The configuration for log processing is specified by the output parameter.

Note

- The configuration file must be encoded in UTF-8 without a byte order mark (BOM). You can use a text editor to modify the file encoding format.
- The path parameter specifies the path to a configuration file. If you configure this parameter, you must use delimiters in the UNIX format. Example: *C:/test/multiline/*.log*. Otherwise, fuzzy match is not supported.
- The values of the type parameters in a configuration file must be the same. If multiple Logstash configuration files are created for a server, the values of the type parameters in the files must be the same.

```
input {
  file {
    type => "iis_log_1"
    path => ["C:/inetpub/logs/LogFiles/W3SVC1/*.log"]
    start_position => "beginning"
  }
}
filter {
  if [type] == "iis_log_1" {
  #ignore log comments
  if [message] =~ "^#" {
    drop {}
  }
  grok {
    # check that fields match your IIS log settings
    match => ["message", "%{TIMESTAMP_ISO8601:log_timestamp} %{IPORHOST:site} %{WORD:me
thod} %{URIPATH:page} %{NOTSPACE:querystring} %{NUMBER:port} %{NOTSPACE:username} %{IPO
RHOST:clienthost} %{NOTSPACE:useragent} %{NUMBER:response} %{NUMBER:subresponse} %{NUMB
ER:scstatus} %{NUMBER:time_taken}"]
  }
    date {
    match => [ "log_timestamp", "YYYY-MM-dd HH:mm:ss" ]
      timezone => "Etc/UTC"
  }
  useragent {
    source=> "useragent"
    prefix=> "browser"
  }
  mutate {
    remove_field => [ "log_timestamp"]
  }
  }
}
output {
  if [type] == "iis_log_1" {
  logservice {
        codec => "json"
        endpoint => "***"
        project => "***"
        logstore => "***"
        topic => ""
        source => ""
        access_key_id => "***"
        access_key_secret => "***"
        max_send_retry => 10
        max_buffer_items => 4000
        max_buffer_bytes => 2097152
        max_buffer_seconds => 3
    }
    }
}
```

Parameters in the log processing configuration

Parameter	Required	Description
endpoint	Yes	The endpoint of the Log Service project.
project	Yes	The name of the Log Service project.
logstore	Yes	The name of the Log Service Logstore.
topic	Yes	The topic of logs.
source	Yes	The source of logs. If you do not configure this parameter, the IP address of your server is automatically returned.
access_key_id	Yes	The AccessKey ID of your Alibaba Cloud account.
access_key_secret	Yes	The AccessKey secret of your Alibaba Cloud account.
max_send_retry	Yes	The maximum number of retries that you can perform when a packet fails to be sent to Log Service. Packets that fail to be sent after the retries are dropped. The retry interval is 200 milliseconds.
max_buffer_items	No	The number of logs that are cached in a packet. If you do not configure this parameter, 4,000 logs are cached in a packet by default.
max_buffer_bytes	No	The size of logs that are cached in a packet. Maximum value: 10485760. Unit: bytes. If you do not configure this parameter, 2,097,152 bytes of logs are cached in a packet by default.
max_buffer_seconds	No	The maximum time period for which logs are cached. Unit: seconds. If you do not configure this parameter, logs are cached for up to 3 seconds by default.

3. Restart Logstash.

   For more information, see Start the service.

What's next

Use PowerShell to launch the logstash.bat process. The logstash.bat process runs in the frontground. In most cases, the logstash.bat process is performed to test and debug log collection. After debugging, we recommend that you configure Logstash as a Windows service. You can run Logstash in the background and at startup. For more information, see Configure Logstash as a Windows service.

6.4.3. Configure Logstash as a Windows service

This topic describes how to use Non-Sucking Service Manager (NSSM) to configure Logstash as a Windows service.

Context

After you use PowerShell to launch the logstash.bat process, the logstash process runs in the frontend. In most cases, the process is used to test and debug log collection. After you complete the debugging, we recommend that you configure Logstash as a Windows service. You can run Logstash in the backend and set auto-run at startup for Logstash. You can use NSSM to configure Logstash as a Windows service. For more information about NSSM, visit NSSM documentation.

You can also use NSSM to start, stop, modify, and remove services in the Command Prompt.

Install Logstash as a Windows service

When you install Logstash for the first time, perform this step. Otherwise, skip this step.

You can run one of the following commands to install Logstash as a Windows service:

- 32-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win32\nssm.exe install logstash "C:\logstash-2.2.2-win\bin\logstash.bat" "agent -f C:\logstash-2.2.2-win\conf"
```

- 64-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win64\nssm.exe install logstash "C:\logstash-2.2.2-win\bin\logstash.bat" "agent -f C:\logstash-2.2.2-win\conf"
```

Start the service

You can run one of the following commands to start the service:

- 32-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win32\nssm.exe start logstash
```

- 64-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win64\nssm.exe start logstash
```

Stop the service

You can run one of the following commands to stop the service:

- 32-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win32\nssm.exe stop logstash
```

- 64-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win64\nssm.exe stop logstash
```

Modify the service

You can run one of the following commands to modify the service:

- 32-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win32\nssm.exe edit logstash
```

- 64-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win64\nssm.exe edit logstash
```

Delete the service

You can run one of the following commands to delete the service:

- 32-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win32\nssm.exe remove logstash
```

- 64-bit system

```
C:\logstash-2.2.2-win\nssm-2.24\win64\nssm.exe remove logstash
```

6.4.4. Advanced features

Logstash provides multiple plug-ins to meet personalized requirements, including:

- grok: parses logs into multiple fields by using regular expressions.
- json_lines and json: structurally parse JSON logs.
- date: parses and converts the date and time fields of logs.
- multiline: customizes multi-line logs.
- kv: structurally parses logs of the key-value pair type.

6.4.5. Logstash error handling

This topic describes how to handle the errors that you encounter while you use Logstash to collect logs.

You can handle the errors based on the following suggestions:

- **Error description**: Data is garbled in the Log Service console.

  Solution: Check whether input files are correctly encoded. Logstash supports UTF-8 file encoding by default.

- **Error description**: The Log Service console displays an error.

  Solution: If the Log Service console displays "`io/console not supported; tty will not be manipulated`", ignore it because it has no impact on the features of Log Service.

If other errors occur, we recommend that you search on Google or Logstash forums for additional information.

6.5. Use SDKs to collect logs

Log Service provides SDKs for multiple programming languages, such as .NET, Java, Python, PHP, and C. You can select an SDK based on your business requirements.

Usage notes

The implementation of Log Service SDKs varies based on the programming languages. Each SDK is an encapsulation of Log Service API in different programming languages. The SDKs provide the following common features:

- Encapsulation of Log Service API. Log Service SDKs implement the underlying API request creation and response parsing. The API operations in different programming languages are similar. This simplifies the switchover between different programming languages. For more information, see Interface regulations.
- Automatic digital signatures. You do not need to focus on the digital signature logic of Log Service API. This simplifies the use of Log Service API. For more information, see Request signatures.
- Protocol Buffer-formatted encapsulation. The logs that are collected by Log Service are encapsulated in the Protocol Buffer format. You do not need to focus on the format details. For more information, see Protocol Buffer format.
- Log compression by using the method that is defined in Log Service API. Log Service SDKs for some programming languages allow you to specify whether logs can be written to Log Service in compression mode. By default, the compression mode is used.
- Unified exception handling mechanism. You can use Log Service SDKs to handle exceptions based on the related programming language. For more information, see Exception handling.
- Support for only synchronous requests.

SDKs

The following table provides links to the references and GitHub source code of Log Service SDKs for different programming languages.

Programming language	References	Source code on GitHub
Java	Overview of Log Service SDK for Java	Log Service SDK for Java and Log Service SDK for Java 0.6.0 API
.NET Core	Overview of Log Service SDK for .NET Core	Log Service SDK for .NET Core
.NET	Overview of Log Service SDK for .NET	Log Service SDK for .NET
PHP	Overview of Log Service SDK for PHP	Log Service SDK for PHP
Python	Overview	Log Service SDK for Python and User guide
Node.js	Overview of Log Service SDK for Node.js	Log Service SDK for Node.js
C	Log Service SDK for C	Log Service SDK for C
GO	Overview	Log Service SDK for Go
iOS	Overview of Log Service SDK for iOS	Log Service SDK for iOS and Log Service SDK for Objective-C
Android	Overview of Log Service SDK for Android	Log Service SDK for Android

Programming language	References	Source code on GitHub
C++	Overview of Log Service SDK for C++	Log Service SDK for C++
JavaScript SDK	SDK for JavaScript	None

7.Collect common logs

7.1. Collect Log4j logs

This topic describes how to use LogHub Log4j Appenders or Logtail to collect Log4j logs.

Context

Log4j is an open-source project of Apache. Log4j allows you to set the output destination and format of logs. The severity levels of logs are classified into ERROR, WARN, INFO, and DEBUG in descending order. The output destination specifies whether logs are sent to the console or files. The output format specifies the format of logs. The following configurations are the default configurations of Log4j:

```
<Configuration status="WARN">
  <Appenders>
    <Console name="Console" target="SYSTEM_OUT">
      <PatternLayout pattern="%d{yyyy-MM-dd HH:mm:ss:SSS zzz} [%t] %-5level %logger{36} - %
msg%n"/>
    </Console>
  </Appenders>
  <Loggers>
    <Logger name="com.foo.Bar" level="trace">
      <AppenderRef ref="Console"/>
    </Logger>
    <Root level="error">
      <AppenderRef ref="Console"/>
    </Root>
  </Loggers>
</Configuration>
```

The following example shows a sample log entry:

```
2013-12-25 19:57:06,954 [10.10.10.10] WARN impl.PermanentTairDaoImpl - Fail to Read Permane
nt Tair,key:e:470217319319741_1,result:com.example.tair.Result@172e3ebc[rc=code=-1, msg=con
nection error or timeout,value=,flag=0]
```

Collect Log4j logs by using LogHub Log4j Appenders

For information about how to collect Log4j logs by using LogHub Log4j Appenders, see Log4j Appender.

Use Logtail to collect Log4j logs

1.
2. On the page that appears, click **RegEx - Text Log** in the **Import Data** section.
3.
4.
5.
6. In the **Logtail Config** step, create a Logtail configuration file.

Parameter	Description
Config Name	The name of the Logtail configuration file. The name cannot be modified after the Logtail configuration file is created. You can also click **Import Other Configuration** to import Logtail configurations from another project.
Log Path	The directories and files from which log data is collected.
Blacklist	If you turn on this switch, you can configure a blacklist in the **Add Blacklist** field. You can configure a blacklist to skip the specified directories or files during log data collection. You can use exact match or wildcard match to specify directories and files. Example:
Docker File	If you collect logs from Docker containers, you can configure the paths and tags of the containers. Logtail monitors the creation and destruction of the containers, filters the logs of the containers by tag, and collects the filtered logs. For more information, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Set the value to **Full Regex Mode**.
Singleline	Turn off the **Singleline** switch.
Log Sample	Enter the following sample log entry in the Log Sample field: `2013-12-25 19:57:06,954 [10.10.10.10] WARN impl.PermanentTairDaoImpl - Fail to Read Permanent Tair,key:e:470217319319741_1,result:com.example.tair.Result@172e3ebc[rc=code=-1, msg=connection error or timeout,value=,flag=0]`
Regex to Match First Line	After you enter the sample log entry, click **Auto Generate**. A regular expression is generated to match the first line of the log entry. The sample log entry starts with a timestamp. Therefore, the generated regular expression is \d+-\d+-\d+\s. *.
Extract Field	If you turn on the **Extract Field** switch, you can use a regular expression to extract field values from logs.

Parameter	Description
RegEx	Set the value to (\d+-\d+-\d+\s\d+:\d+:\d+,\d+)\s\[([^\]]*)\]\s(\S+)\s+(\S+)\s-\s(. *). You can configure a regular expression based on one of the following methods: ○ Automatically generate a regular expression In the **Log Sample** field, select the field values to be extracted, and click **Generate Regular Expression**. A regular expression is automatically generated. ○ Manually enter a regular expression Click **Manual**. In the RegEx field, enter a regular expression. After you enter a regular expression in the field, click **Validate** to check whether the regular expression can parse the log content. For more information, see How do I modify a regular expression?.
Extracted Content	After you use a regular expression to extract field values, you must specify a key for each value.
Use System Time	Turn off the **Use System Time** switch. Configure the time field in the %Y-%m-%dT%H:%M:%S format. You can use one of the following methods to configure the time field: ○ If you turn on **Use System Time**, the timestamp of a log indicates the system time when the log is collected. The system time refers to the time of the server or container on which Logtail runs. ○ If you turn off **Use System Time**, you must specify the time field for the **Extracted Content** parameter and configure **Time Conversion Format** based on the value of the time field. For more information about the time format, see Time formats.
Drop Failed to Parse Logs	○ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ○ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	The maximum depth at which the specified log directory is monitored. Valid values: 0 to 1000. The value 0 indicates that only the directory that is specified in the log path is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ○ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ○ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ○ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ○ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ○ Custom: If you select this value, you must select a time zone based on your business requirements.
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ○ Never: All log files are continuously monitored and never time out. ○ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.

Parameter	Description
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

After you complete the Logtail configurations, Log Service starts to collect Log4j logs.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

> **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

7.2. Collect Python logs

This topic describes how to use Logtail to collect Python logs.

Context

The logging module of Python is a logging system that is compatible with third-party modules or applications. The logging module defines multiple log severity levels and logging methods. The logging module consists of four components: loggers, handlers, filters, and formatters.

Formatters specify the output format of logs. The fields in the configurations of a formatter are in the %(key)s format.

```
import logging
import logging.handlers
LOG_FILE = 'tst.log'
handler = logging.handlers.RotatingFileHandler(LOG_FILE, maxBytes = 1024*1024, backupCount
= 5) # Create a handler object.
%(asctime)s - %(filename)s:%(lineno)s - %(levelno)s %(levelname)s %(pathname)s %(module)s %
(funcName)s %(created)f %(thread)d %(threadName)s %(process)d %(name)s - %(message)s // Def
ine the output format of logs.
formatter = logging.Formatter(fmt)   # Create a formatter object.
handler.setFormatter(formatter)      # Add the formatter to the handler.
logger = logging.getLogger('tst')    # Retrieve a logger that is named tst.
logger.addHandler(handler)           # Add the handler to the logger.
logger.setLevel(logging.DEBUG)
logger.info('first info message')
logger.debug('first debug message')
```

The following table describes the fields in the formatter configurations.

Field	Description
%(name)s	The name of the logger that generates a log.
%(levelno)s	The severity level of a log in the numeric format. Valid values: 10, 20, 30, 40, and 50.
%(levelname)s	The severity level of a log in the text format. Valid values: DEBUG, INFO, WARNING, ERROR, and CRITICAL.
%(pathname)s	The full path name of the source file where the logging call is initiated.
%(filename)s	The name of the source file.
%(module)s	The name of the module where the logging call is initiated.
%(funcName)s	The name of the function from which the logging call is initiated.

Field	Description
%(lineno)d	The line number in the source file where the logging call is initiated.
%(created)f	The time when a log is created. The value is a Unix timestamp. It represents the number of seconds that have elapsed since January 1, 1970, 00:00:00 (UTC).
%(relativeCreated)d	The difference between the time when a log is created and the time when the logging module is loaded. Unit: milliseconds.
%(asctime)s	The time when a log is created. Example: 2003-07-08 16:49:45,896. The digits after the comma (,) indicate the millisecond portion of the time.
%(msecs)d	The millisecond portion of the time when a log is created.
%(thread)d	The ID of the thread.
%(threadName)s	The name of the thread.
%(process)d	The ID of the process.
%(message)s	The log content.

The following example shows sample log entries:

```
2015-03-04 23:21:59,682 - log_test.py:16 - tst - first info message
2015-03-04 23:21:59,682 - log_test.py:17 - tst - first debug message
```

Procedure

1.
2. On the page that appears, click **RegEx - Text Log** in the **Import Data** section.
3.
4.
5.
6. In the **Logtail Config** step, create a Logtail configuration file.

Parameter	Description
Config Name	The name of the Logtail configuration file. The name cannot be modified after the Logtail configuration file is created. You can also click **Import Other Configuration** to import a Logtail configuration file from another project.
Log Path	The directories and files from which log data is collected.

Parameter	Description
Blacklist	If you turn on this switch, you can configure a blacklist in the **Add Blacklist** field. You can configure a blacklist to skip the specified directories or files during log data collection. You can use exact match or wildcard match to specify directories and files. Example:
Docker File	If you collect logs from Docker containers, you can configure the paths and tags of the containers. Logtail monitors the creation and destruction of the containers, filters the logs of the containers by tag, and collects the filtered logs. For more information, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Set the value to **Full Regex Mode**.
Singleline	Turn on the **Singleline** switch. The single-line mode indicates that each line contains one log entry.
Log Sample	Enter the following sample log entry in the Log Sample field: `2016-02-19 11:06:52,514 - test.py:19 - 10 DEBUG test.py test <module> 1455851212.514271 139865996687072 MainThread 20193 tst - first debug message`
Extract Field	If you turn on the **Extract Field** switch, you can use a regular expression to extract field values from logs.
RegEx	Set the value to (\d+-\d+-\d+\s\S+)\s-\s([^:]+):(\d+)\s+-\s+(\d+)\s+(\w+)\s+(\S+)\s+(\w+)\s+(\S+)\s+(\S+)\s+(\d+)\s+(\w+)\s+(\d+)\s+(\w+)\s+-\s+(. *). You can configure a regular expression based on one of the following methods: ○ Automatically generate a regular expression In the **Log Sample** field, select the field values to be extracted, and click **Generate Regular Expression**. A regular expression is automatically generated. ○ Manually enter a regular expression Click **Manual**. In the RegEx field, enter a regular expression. After you enter a regular expression in the field, click **Validate** to check whether the regular expression can parse the log content. For more information, see How do I modify a regular expression?.
Extracted Content	The field is available only after you turn on the **Extract Field** switch. After you use a regular expression to extract field values, you must specify a key for each value.
Use System Time	The field is available only after you turn on the **Extract Field** switch.
Drop Failed to Parse Logs	

Parameter	Description
Maximum Directory Monitoring Depth	The maximum depth at which the specified log directory is monitored. Valid values: 0 to 1000. The value 0 indicates that only the directory that is specified in the log path is monitored.

After you complete the Logtail configurations, Log Service starts to collect Python logs.

7.

7.3. Collect Node.js logs

This topic describes how to use Logtail to collect Node.js logs.

Context

Log4js is a management tool for Node.js logs. You can use Log4js to send Node.js logs to files and customize the log format.

```
var log4js = require('log4js');
log4js.configure({
  appenders: [
    {
      type: 'file', //Send logs to a file.
      filename: 'logs/access.log',
      maxLogSize: 1024,
      backups:3,
      category: 'normal'
    }
  ]
});
var logger = log4js.getLogger('normal');
logger.setLevel('INFO');
logger.info("this is a info msg");
logger.error("this is a err msg");
```

The preceding code shows how to configure Log4js to send logs to a file. Logs that are sent by Log4js are classified into six severity levels. The severity levels are TRACE, DEBUG, INFO, WARN, ERROR, and FATAL in ascending order. The following example shows two sample log entries:

```
[2016-02-24 17:42:38.946] [INFO] normal - this is a info msg
[2016-02-24 17:42:38.951] [ERROR] normal - this is a err msg
```

Procedure

1.
2. On the page that appears, click **RegEx - Text Log** in the **Import Data** section.
3.
4.
5.
6. In the **Logtail Config** step, create a Logtail configuration file.

Parameter	Description
Config Name	The name of the Logtail configuration file. The name cannot be modified after the Logtail configuration file is created. You can also click **Import Other Configuration** to import a Logtail configuration file from another project.
Log Path	The directories and files from which log data is collected.
Blacklist	If you turn on this switch, you can configure a blacklist in the **Add Blacklist** field. You can configure a blacklist to skip the specified directories or files during log collection. You can use exact match or wildcard match to specify directories and files. Example:
Docker File	If you collect logs from Docker containers, you can configure the paths and tags of the containers. Logtail monitors the creation and destruction of the containers, filters the logs of the containers by tag, and collects the filtered logs. For more information, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Set the value to **Full Regex Mode**.
Singleline	Turn on the **Singleline** switch. The single-line mode indicates that each line contains one log entry.
Log Sample	Enter the following sample log entry in the Log Sample field: `[2016-01-31 12:02:25.844] [INFO] access - 10.10.10.10 - - "GET /user/projects/ali_sls_log? ignoreError=true HTTP/1.1" 304 - "http://aliyun.com/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/10.10.10.10 Safari/537.36"`
Extract Field	If you turn on the **Extract Field** switch, you can use a regular expression to extract field values from logs.
RegEx	Set the value to \[([^]]+)]\s\[(\w+)]\s(\w+)\s-\s(\S+)\s-\s-\s"([^"]+)"\s(\d+)[^"]+("[^"]+)"\s"([^"]+). *. You can configure a regular expression based on one of the following methods: ○ Automatically generate a regular expression In the **Log Sample** field, select the field values to be extracted, and click **Generate Regular Expression**. A regular expression is automatically generated. ○ Manually enter a regular expression Click **Manual**. In the RegEx field, enter a regular expression. After you enter a regular expression in the field, click **Validate** to check whether the regular expression can parse the log content. For more information, see How do I modify a regular expression?.

Parameter	Description
Extracted Content	After you use a regular expression to extract field values, you must specify a key for each value.
Use System Time	The field is available only after you turn on the **Extract Field** switch.
Drop Failed to Parse Logs	
Maximum Directory Monitoring Depth	The maximum depth at which the specified log directory is monitored. Valid values: 0 to 1000. The value 0 indicates that only the directory that is specified in the log path is monitored.

After you complete the Logtail configurations, Log Service starts to collect Node.js logs.

7.

7.4. Collect WordPress logs

This topic describes how to use Logtail to collect WordPress logs.

Context

WordPress is a blog platform that is developed in the PHP programming language and paired with a MySQL database. WordPress has evolved into a software application for content management. The following example shows a sample log entry:

```
10.10.10.10 - - [07/Jan/2016:21:06:39 +0800] "GET /wp-admin/js/password-strength-meter.min.
js? ver=4.4 HTTP/1.0" 200 776 "http://wordpress.c4a1a0aecdb1943169555231dcc4adfb7.cn-hangzh
ou.alicontainer.com/wp-admin/install.php" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/10.10.10.10 Safari/537.36"
```

Procedure

1.
2. On the page that appears, click **RegEx - Text Log** in the **Import Data** section.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   - If a machine group is available, click **Use 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 an ECS instance that belongs to a different Alibaba Cloud account, a server in an on-premises data center, or a server of a third-party cloud service provider, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server. After you manually install Logtail, you must configure a user identifier on the server. For more information, see Configure a user identifier.

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 **Logtail Config** step, create a Logtail configuration file.

Parameter	Description
Config Name	The name of the Logtail configuration file. The name cannot be modified after the Logtail configuration file is created. You can also click **Import Other Configuration** to import a Logtail configuration file from another project.
Log Path	The directories and files from which log data is collected.

Parameter	Description
Blacklist	If you turn on this switch, you can configure a blacklist in the **Add Blacklist** field. You can configure a blacklist to skip the specified directories or files during log data collection. You can use exact match or wildcard match to specify directories and files. Example: ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir1* for Content, all files in the */home/admin/dir1* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/dir** for Content, the files in all subdirectories whose names are prefixed by dir in the */home/admin/* directory are skipped. ◦ If you select **Filter by Directory** from a drop-down list in the Filter Type column and enter */home/admin/*/dir* for Content, all files in dir directories in each subdirectory of the */home/admin/* directory are skipped. For example, the files in the */home/admin/a/dir* directory are skipped, but the files in the */home/admin/a/b/dir* directory are not skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*.log* for Content, all files whose names are prefixed by private and suffixed by .log in the */home/admin/* directory are skipped. ◦ If you select **Filter by File** from a drop-down list in the Filter Type column and enter */home/admin/private*/*_inner.log* for Content, all files whose names are suffixed by _inner.log in the subdirectories whose names are prefixed by private in the */home/admin/* directory are skipped. For example, the */home/admin/private/app_inner.log* file is skipped, but the */home/admin/private/app.log* file is not skipped.
Docker File	If you collect logs from Docker containers, you can configure the paths and tags of the containers. Logtail monitors the creation and destruction of the containers, filters the logs of the containers by tag, and collects the filtered logs. For more information, see Use the Log Service console to collect container text logs in DaemonSet mode.
Mode	Set the value to **Full Regex Mode**.
Singleline	Turn off the **Singleline** switch.
Log Sample	Enter the following sample log entry in the Log Sample field: `10.10.10.10 - - [07/Jan/2016:21:06:39 +0800] "GET /wp-admin/js/password-strength-meter.min.js? ver=4.4 HTTP/1.0" 200 776 "http://wordpress.c4a1a0aecdb1943169555231dcc4adfb7.cn-hangzhou.alicontainer.com/wp-admin/install.php" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/10.10.10.10 Safari/537.36"`

Parameter	Description
Regex to Match First Line	After you enter the sample log entry, click **Auto Generate**. A regular expression is generated to match the first line of a log entry. The sample log entry starts with an IP address. Therefore, the generated regular expression is \d+\.\d+\.\d+\.\d+\s-\s. *.
Extract Field	If you turn on the **Extract Field** switch, you can use a regular expression to extract field values from logs.
RegEx	Set the value to (\S+) - - \[([^\]]*)] "(\S+) ([^"]+)" (\S+) (\S+) "([^"]+)" "([^"]+)". You can configure a regular expression based on one of the following methods: ○ Automatically generate a regular expression In the **Log Sample** field, select the field values to be extracted, and click **Generate Regular Expression**. A regular expression is automatically generated. ○ Manually enter a regular expression Click **Manual**. In the RegEx field, enter a regular expression. After you enter a regular expression in the field, click **Validate** to check whether the regular expression can parse the sample log entry. For more information, see How do I modify a regular expression?.
Extracted Content	The field is available only after you turn on the **Extract Field** switch. After you use a regular expression to extract field values, you must specify a key for each value.
Use System Time	Turn off the **Use System Time** switch. Configure the time field in the %d/%b/%Y:%H:%M:%S format. You can use one of the following methods to configure the time field: ○ If you turn on **Use System Time**, the timestamp of a log indicates the system time when the log is collected. The system time refers to the time of the server or container on which Logtail runs. ○ If you turn off **Use System Time**, you must specify the time field for the **Extracted Content** parameter and configure **Time Conversion Format** based on the value of the time field. For more information about the time format, see Time formats.
Drop Failed to Parse Logs	○ If you turn on **Drop Failed to Parse Logs**, the logs that fail to be parsed are not uploaded to Log Service. ○ If you turn off **Drop Failed to Parse Logs**, the logs that fail to be parsed are still uploaded to Log Service as the value of the __raw_log__ field.
Maximum Directory Monitoring Depth	The maximum depth at which the specified log directory is monitored. Valid values: 0 to 1000. The value 0 indicates that only the directory that is specified in the log path is monitored.

You can configure advanced settings based on your business requirements. We recommend that you do not modify the advanced settings. The following table describes the parameters in the advanced settings.

Parameter	Description
Enable Plug-in Processing	If you turn on **Enable Plug-in Processing**, you can configure Logtail plug-ins to process logs. For more information, see Overview. **Note** If you turn on **Enable Plug-in Processing**, the parameters such as Upload Raw Log, Timezone, Drop Failed to Parse Logs, Filter Configuration, and Incomplete Entry Upload (Delimiter mode) become unavailable.
Upload Raw Log	If you turn on **Upload Raw Log**, each raw log is uploaded to Log Service as the value of the __raw__ field together with the log parsed from the raw log.
Topic Generation Mode	Select the topic generation mode. For more information, see Log topics. ◦ **Null - Do not generate topic**: In this mode, the topic field is set to an empty string. When you query logs, you do not need to specify a topic. This is the default value. ◦ **Machine Group Topic Attributes**: In this mode, topics are configured at the machine group level. If you want to distinguish the logs that are generated by different servers, select this mode. ◦ **File Path RegEx**: In this mode, you must specify a regular expression in the **Custom RegEx** field. The part of a log path that matches the regular expression is used as the topic. If you want to distinguish the logs that are generated by different users or instances, select this mode.
Log File Encoding	Select the encoding format of log files. Valid values: utf8 and gbk.
Timezone	Select the time zone in which logs are collected. Valid values: ◦ System Timezone: If you select this value, the time zone of the server or the container on which Logtail is installed is used. ◦ Custom: If you select this value, you must select a time zone based on your business requirements.
Timeout	Select a timeout period of log files. If a log file is not updated within the specified period, Logtail considers the file to be timed out. Valid values: ◦ Never: All log files are continuously monitored and never time out. ◦ 30 Minute Timeout: If a log file is not updated within 30 minutes, Logtail considers the file to be timed out and stops monitoring the file. If you select **30 Minute Timeout**, you must configure the **Maximum Timeout Directory Depth** parameter. Valid values: 1 to 3.

Parameter	Description
Filter Configuration	Specify the filter conditions that you want to use to collect logs. Only the logs that match the specified filter conditions are collected. Examples: ○ Collect the logs that match the specified filter conditions: If you set **Key** to **level** and **RegEx** to **WARNING|ERROR**, only the logs whose level is WARNING or ERROR are collected. ○ Filter out the logs that do not match the specified filter conditions. For more information, see Regular-Expressions.info. ■ If you set **Key** to **level** and **RegEx** to **^(?!.*(INFO|DEBUG)).***, the logs whose level contains INFO or DEBUG are not collected. ■ If you set **Key** to **level** and **RegEx** to **^(?!(INFO|DEBUG)$).***, the logs whose level is INFO or DEBUG are not collected. ■ If you set **Key** to **url** and **RegEx** to **.*^(?!.*(healthcheck)).***, the logs whose url contains healthcheck are not collected. For example, if a log has the Key field of url and the Value field of /inner/healthcheck/jiankong.html, the log is not collected. For more information, see regex-exclude-word and regex-exclude-pattern.
First Collection Size	Specify the size of data that Logtail can collect from a log file the first time Logtail collects logs from the file. The default value of **First Collection Size** is 1024. Unit: KB. ○ If the file size is less than 1,024 KB, Logtail collects data from the beginning of the file. ○ If the file size is greater than 1,024 KB, Logtail collects the last 1,024 KB of data in the file. You can specify **First Collection Size** based on your business requirements. Valid values: 0 to 10485760. Unit: KB.
More Configurations	Specify extended settings for Logtail. For more information, see advanced. For example, if you want to use the current Logtail configuration to collect logs from log files that match a different Logtail configuration and specify the interval at which logs are aggregated and sent to Log Service, you can specify extended settings for the current Logtail. `{ "force_multiconfig": true, "batch_send_interval": 3 }`

After you complete the Logtail configurations, Log Service starts to collect WordPress logs.

7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

> **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

7.5. Collect Unity3D logs

This topic describes how to use the web tracking feature of Log Service to collect Unity3D logs.

Context

Unity3D is a cross-platform game engine that is developed by Unity Technologies. The engine allows you to create 3D video games, VR buildings, real-time 3D animation, and other interactive content.

Log Service allows you to use the web tracking feature to collect Unity3D logs. For more information about the web tracking feature, see Use web tracking to collect logs. Unity Debug.Log is used as an example to describe how to collect Unity3D logs.

Procedure

1. Enable the web tracking feature. For more information, see Use web tracking to collect logs.
2. Create a Unity3D logging handler.

   In the Unity editor, create a C# file that is named *LogOutputHandler.cs*, add the following code to the file, and modify the following variables:

   - project: the name of the Log Service project.
   - logstore: the name of the Logstore.
   - serviceAddr: the endpoint of the Log Service project. For more information, see Endpoints.

```
using UnityEngine;
using System.Collections;
public class LogOutputHandler : MonoBehaviour
{
    //Register the HandleLog function on scene start to fire on debug.log events
    public void OnEnable()
    {
        Application.logMessageReceived += HandleLog;
    }
    //Remove callback when object goes out of scope
    public void OnDisable()
    {
        Application.logMessageReceived -= HandleLog;
    }
    string project = "your project name";
    string logstore = "your logstore name";
    string serviceAddr = "http address of your log service project";
    //Capture debug.log output, send logs to Loggly
    public void HandleLog(string logString, string stackTrace, LogType type)
    {
        string parameters = "";
        parameters += "Level=" + WWW.EscapeURL(type.ToString());
        parameters += "&";
        parameters += "Message=" + WWW.EscapeURL(logString);
        parameters += "&";
        parameters += "Stack_Trace=" + WWW.EscapeURL(stackTrace);
        parameters += "&";
        //Add any User, Game, or Device MetaData that would be useful to finding issues
later
        parameters += "Device_Model=" + WWW.EscapeURL(SystemInfo.deviceModel);
        string url = "http://" + project + "." + serviceAddr + "/logstores/" + logstore
+ "/track? APIVersion=0.6.0&" + parameters;
        StartCoroutine(SendData(url));
    }
    public IEnumerator SendData(string url)
    {
        WWW sendLog = new WWW(url);
        yield return sendLog;
    }
}
```

You can use the preceding script to asynchronously send logs to Log Service. You can also specify other fields in the script to collect the fields.

3. Generate Unity3D logs.

   Create a file that is named *LogglyTest.cs*, and add the following code to the file:

```
using UnityEngine;
using System.Collections;
public class LogglyTest : MonoBehaviour {
    void Start () {
        Debug.Log ("Hello world");
    }
}
```

4. View collected log data.

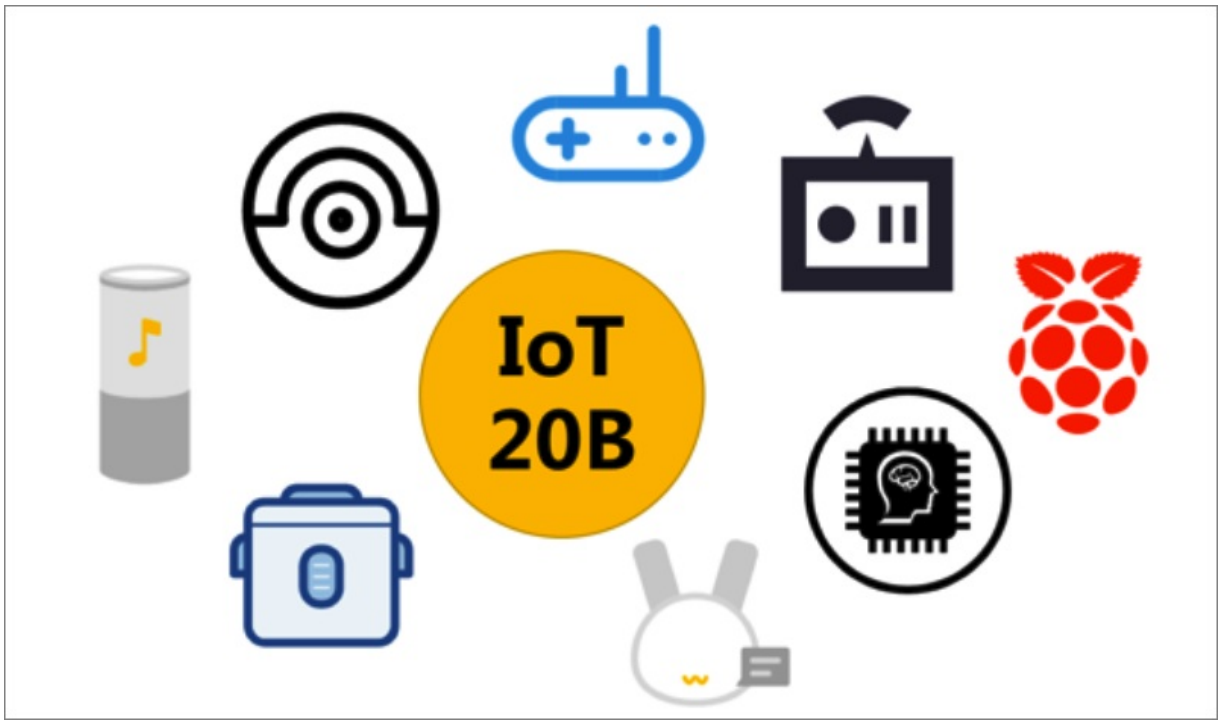
   After you run the Unity3D application, logs are generated and sent to Log Service. You can view the logs in the Log Service console.

8.Best practices

8.1. Collect IoT or embedded development logs

The global market for Internet of Things (IoT) solutions is growing fast. More and more intelligent IoT devices are applied to daily life, such as smart routers, TV dongles, Tmall Genie, and robot vacuum cleaners. However, the embedded development model in the traditional software industry poses big challenges to the IoT industry. For example, traditional device logging solutions cannot meet requirements because a large number of IoT devices are widely distributed in different places and are difficult to debug or scale.

To address these challenges, Log Service provides C Producer Library based on years of experience in developing Logtail. C Producer Library is a log collection solution that is customized for IoT devices.

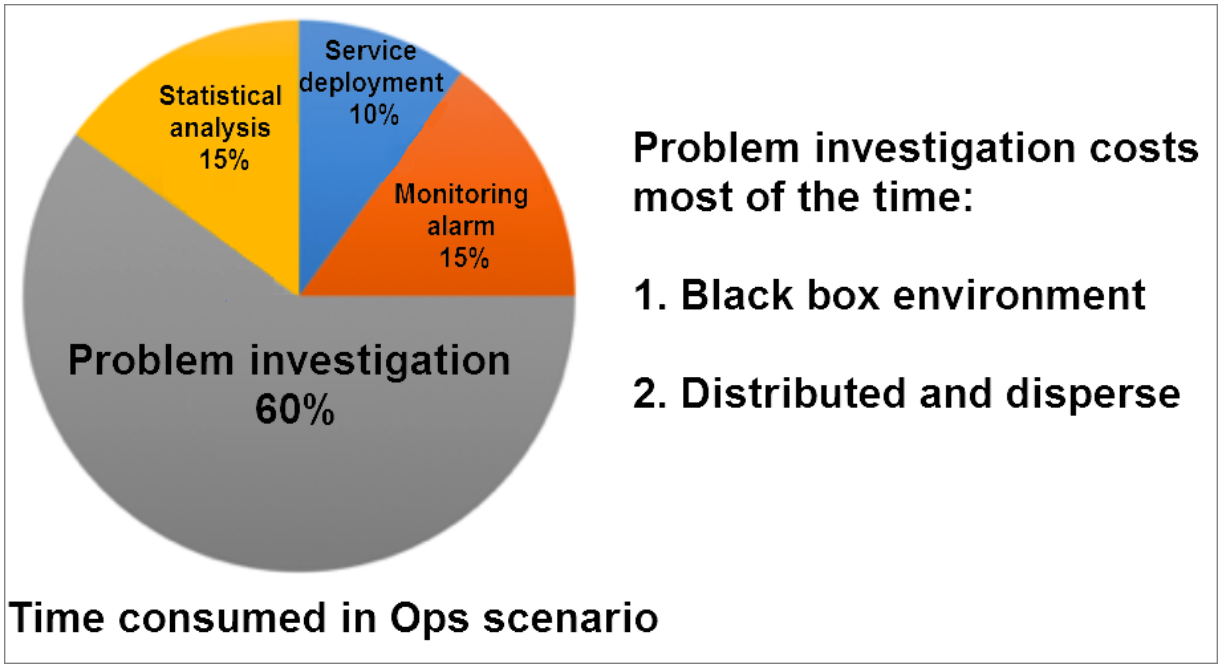


Requirements in embedded software development

IoT or embedded software developers must have profound knowledge and rich experience to manage, monitor, and troubleshoot a large number of IoT devices. Embedded software development has the following requirements:

- Data collection: Collect data from millions or tens of millions of devices distributed around the world in real time.
- Debugging: Use a set of solutions to collect data online and debug software in real time.
- Online troubleshooting: Locate an online IoT device and identify the error cause if an error occurs on the device.
- Monitoring: Monitor the number, statuses, and locations of online IoT devices, and configure alert rules for the devices.

- Real-time data analysis: Visualize the data generated by IoT devices to create user profiles by connecting the devices with real-time computing platforms and big data warehouses.



Major challenges in the IoT field

Traditional software development solutions cannot meet the preceding requirements in the IoT industry due to the following challenges:

- A large number of IoT devices are deployed. In traditional O&M, managing 10,000 servers is a big challenge. However, in the IoT industry, managing 100,000 online IoT devices is only a basic requirement.
- IoT devices are widely distributed in different places within a country or around the world.
- IoT devices are in unknown statuses in most cases and are difficult to access and debug.
- IoT devices are equipped with low-end hardware to reduce costs. For example, an IoT device may have a total memory size of 32 MB. As a result, traditional log collection solutions for PCs do not apply to IoT devices.

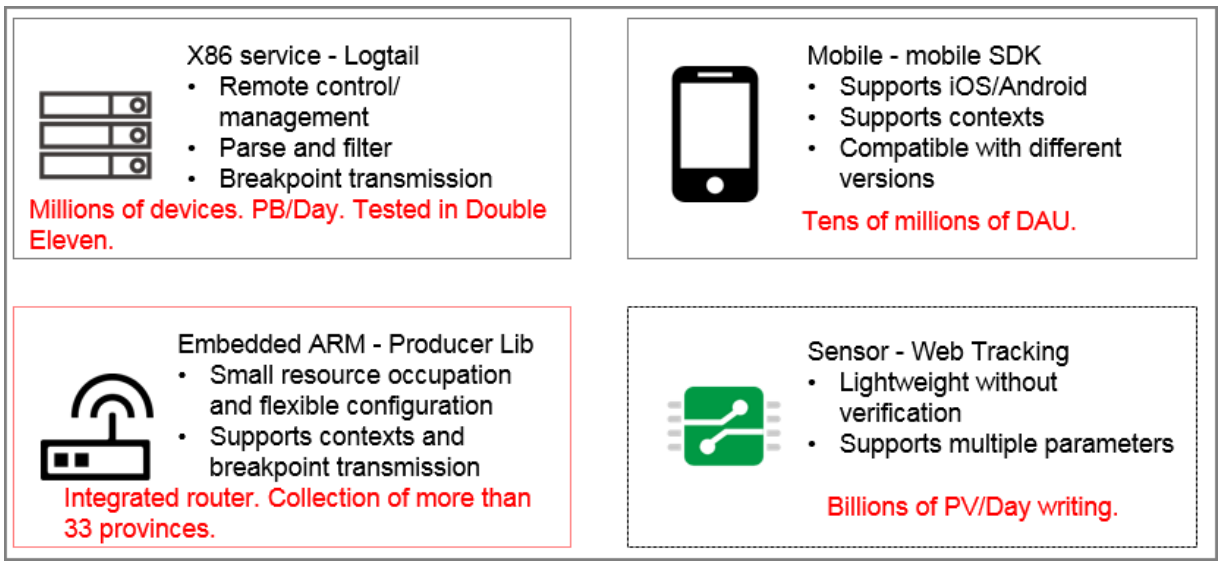
C Producer

A log data collection solution customized by Log Service

Logtail is the agent of Log Service and is deployed on millions of x86 servers. In addition, Log Service provides a variety of collection solutions:

- Mobile SDK: You can use the SDKs to collect data from Android or iOS platforms with tens of millions of daily active users (DAUs).
- Web tracking (JavaScript): Web tracking is a lightweight solution that is similar to Baidu Tongji and Google Analytics. You can use web tracking to collect data without a signature.

In these solutions, C Producer Library is a solution that is customized based on Logtail to collect log data from IoT devices. The solution is compatible with the CPU, memory, disk, network, and application mode of IoT devices. The following figure shows the features of these solutions.



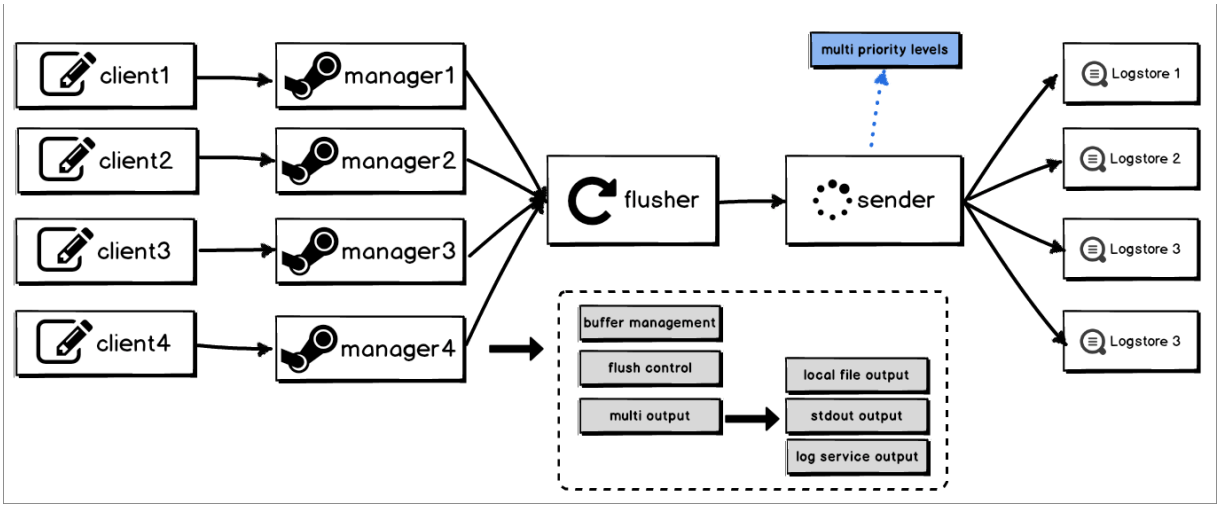
Features of C Producer Library

C Producer Library serves as a lightweight Logtail to offer high stability, high performance, and low resource consumption. Compared with Logtail, C Producer Library does not support real-time configuration management. However, C Producer Library inherits the following features of Logtail:

- Multiple tenants: C Producer Library can process multiple types of logs such as Metric, DebugLog, and ErrorLog based on their priorities. You can configure multiple clients for C Producer Library. You can also configure a collection priority, destination project, and destination Logstore for each client.
- Contextual query: Logs generated by the same client are in the same context. You can view the relevant logs before and after a specified log.
- Concurrent sending and resumable upload: You can set the maximum cache size. If the size of cached logs reaches the upper limit, no more logs can be written to the cache.

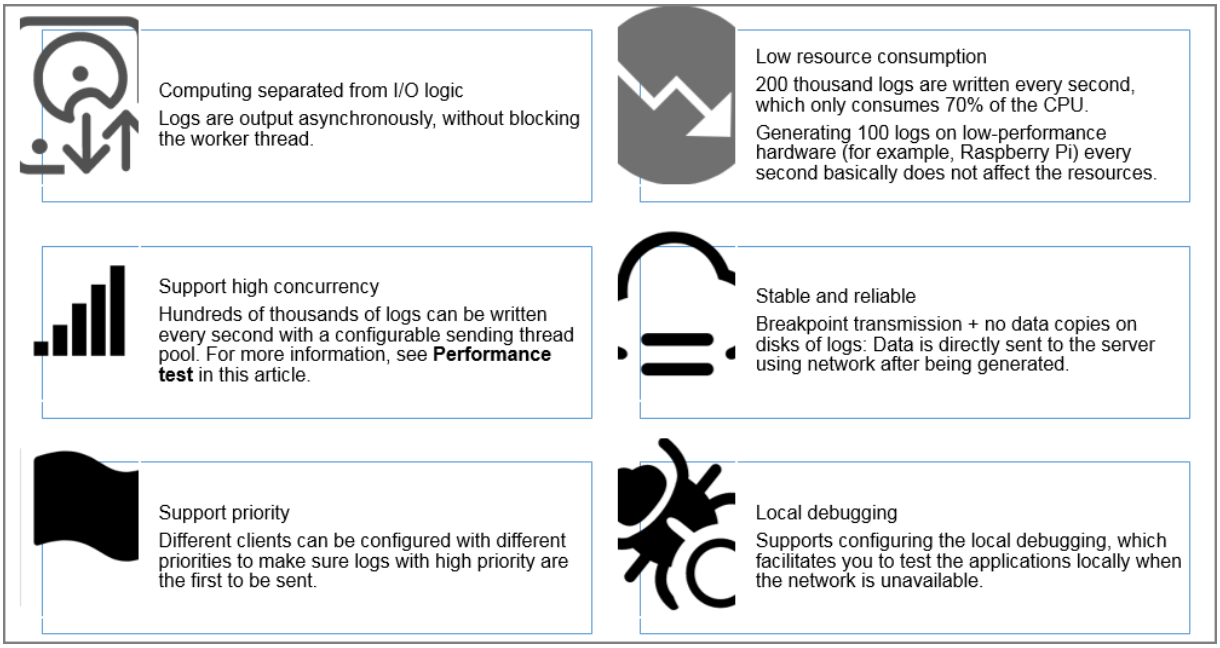
In addition, C Producer Library provides the following features that are specific to IoT devices:

- Local debugging: You can export logs to local devices. You can configure the log rotation, log quantity, and rotation size.
- Fine-grained resource control: You can set cache sizes and aggregate modes for different types of data or logs.
- Log cache compression: If log data fails to be sent to Logstores, the log data can be compressed to reduce the memory usage of IoT devices.



Advantages of C Producer Library

As a custom solution for IoT devices, C Producer Library has the following advantages:



- Highly concurrent write traffic: You can configure the thread pool of clients to write hundreds of thousands of logs per second to Log Service. For more information, see the "Performance test" section in this topic.
- Low resource consumption: Only 70% of the CPU resources are required to write 200,000 logs per second. For low-performance hardware such as Raspberry Pi, the CPU resources are not affected even if 100 logs are generated per second.
- Direct log shipping: After logs are generated, they are directly sent to Log Service.
- Logic isolation between computing and I/O: Logs are generated in an asynchronous manner without blocking worker threads.
- Multiple priorities: You can configure different priorities for your clients to ensure that logs with higher priorities are sent first.
- Local debugging: You can configure local debugging to test your applications if the network connection is unavailable.

C Producer Library simplifies application development. You do not need to consider the implementation of log collection or the impact on your business. This makes data collection much easier.

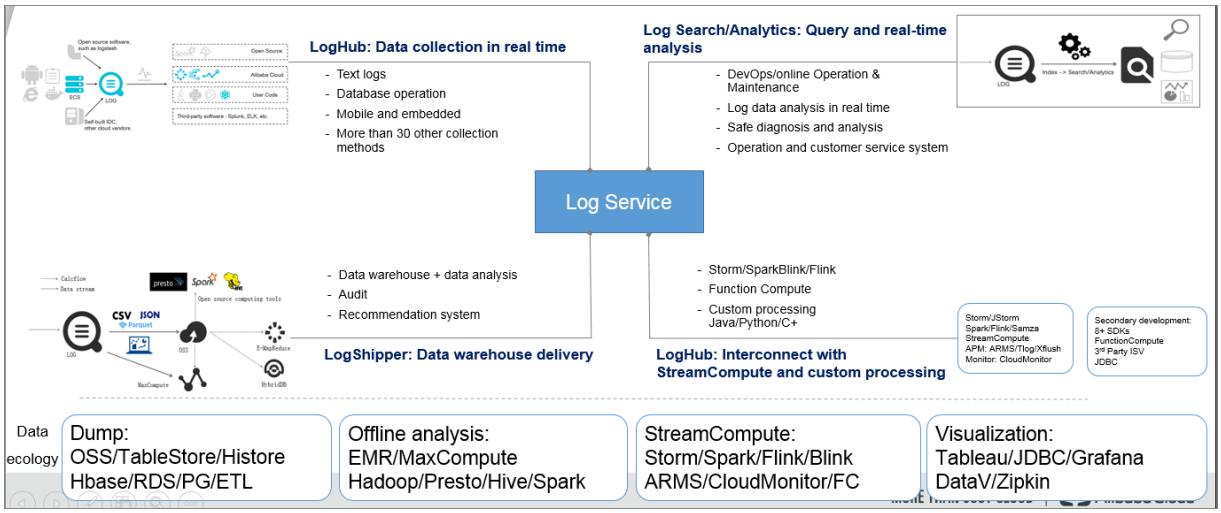
The following table compares C Producer Library and other embedded collection solutions.

Type		C Producer	Other solutions
Programming	Platform	Mobile-based and embedded	Mobile-based
	Context	Supported	Not supported
	Multiple log types	Supported	Not supported
	Custom Format	Supported	Not supported (Several limited fields are provided)
	Priority	Supported	Not supported
	Environment parameter	Configurable	Configurable
Stability	Concurrency	High	Medium
	Compression algorithm	LZ4 (balance between efficiency and performance) and Gzip	Optimized
	Low resource consumption	Optimized	Medium
Transmission	Resumable upload	Supported	By default, resumable upload is not supported. Secondary development is required to support resumable upload.
	Endpoint	8 endpoints in China and 8 endpoints outside China	Hangzhou
Debugging	Local log	Supported	Supported in manual mode
	Parameter setting	Supported	Not supported
Real-time collection	Visible on the server side	1 second (99.9%) to 3 seconds (maximum)	1 to 2 hours
Custom processing		More than 15 connection modes	Custom real-time and offline solutions

C Producer Library + Log Service solution

C Producer Library can be integrated with Log Service to provide a full range of log collection solutions for IoT devices.

- Large scale
  - Writes hundreds of millions of logs from clients to Log Service in real time.
  - Writes petabytes of data per day.
- High speed
  - Fast collection: Logs can be consumed immediately after being written to Log Service.
  - Fast query: Billions of data records can be processed and queried within 1 second by executing a complex query statement in which you specify five conditions.
  - Fast analysis: Hundreds of millions of data records can be aggregated and analyzed within 1 second by executing a complex analysis statement that includes five aggregate functions and the GROUP BY clause.
- High compatibility
  - Seamlessly integrated with various Alibaba Cloud services.
  - Compatible with various open source storage, computing, and visualization systems.

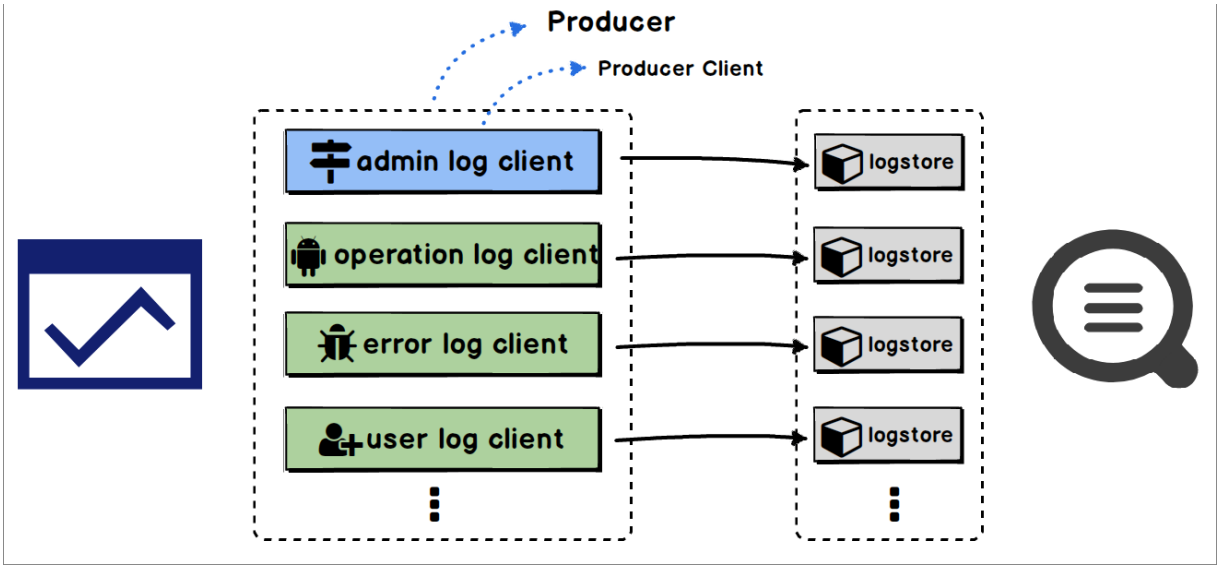


Download and use C Producer Library

Download URL: GitHub

You can create multiple producers for each application and create multiple clients for each producer. You can configure the destination IP address, log level, local debugging, cache size, custom identifier, and topic for each client.

For more information about how to install C Producer Library, see README.



Performance test

Environment configuration

- High-performance scenarios: traditional x86 servers.
- Low-performance scenarios: Raspberry Pi (environments with low power consumption).

The following figure shows the configuration details.

High-performance scenario	Low-performance scenario
• CPU: Intel(R) Xeon(R) CPU E5-2682 v4 @ 2.50 GHz • Memory: 64 GB • Operating system: Linux version 2.6.32-220.23.2.ali1113.el5.x86_64 • GCC: 4.1.2 • C-Producer: Dynamic library is 162 KB. Static library is 140 KB. (Use the static library for test. The binary after compilation is 157 KB. All are stripped.)	Type: Raspberry Pi 3B CPU: Broadcom BCM2837 1.2 GHz A53 64 bit. (Use the host USB for power supply. The frequency is lowered to 600 MHz.) Memory: 1 GB DDR2 Operating system: Linux 4.9.41-v7+ #1023 SMP armv71 GNU/Linux GCC: 6.3.0 (Raspbian 6.3.0-18+rpi1) C-Producer: Dynamic library is 179 KB. Static library is 162 KB. (Use the static library for test. The binary after compilation is 287 KB. All are stripped.)

C Producer Library configuration

- ARM (Raspberry Pi)
  - Cache size: 10 MB
  - Aggregation time: 3 seconds. If the aggregation time, size of the aggregated packet, or number of aggregated logs reaches the specified thresholds, the data is encapsulated and sent to Log Service.
  - Size of the aggregated packet: 1 MB
  - Number of aggregated logs: 1,000
  - Number of sending threads: 1
  - Number of custom tags: 5
- X86

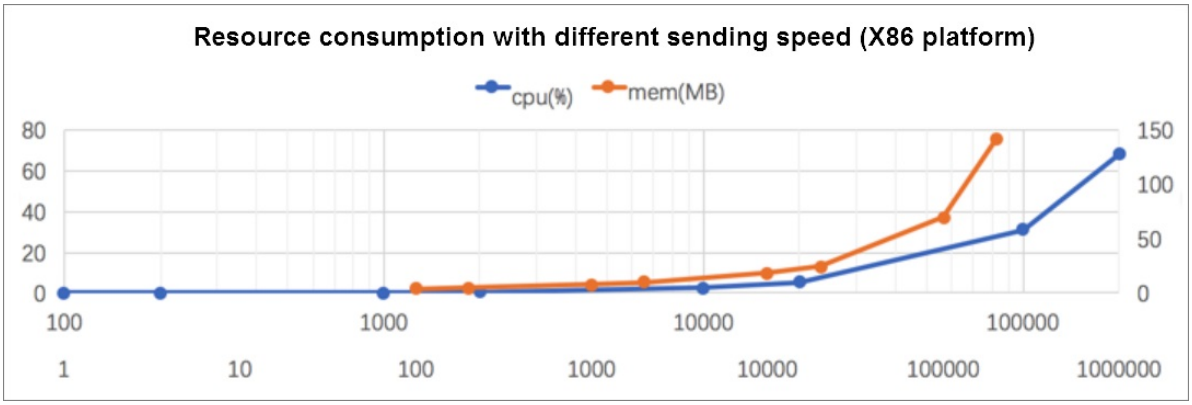
- Cache size: 10 MB
- Aggregation time: 3 seconds. If the aggregation time, size of the aggregated packet, or number of aggregated logs reaches the specified thresholds, the data is encapsulated and sent to Log Service.
- Size of the aggregated packet: 3 MB
- Number of aggregated logs: 4096
- Number of sending threads: 4
- Number of custom tags: 5

The following sample log contains nine key-value pairs:

```
__source__: 192.0.2.1
__tag__:1: 2
__tag__:5: 6
__tag__:a: b
__tag__:c: d
__tag__:tag_key: tag_value
__topic__: topic_test
_file_: /disk1/workspace/tools/aliyun-log-c-sdk/sample/log_producer_sample.c
_function_: log_producer_post_logs
_level_: LOG_PRODUCER_LEVEL_WARN
_line_: 248
_thread_: 40978304
LogHub: Real-time log collection and consumption
Search/Analytics: Query and real-time analysis
Interconnection: Grafana and JDBC/SQL92
Visualized: dashboard and report functions
```

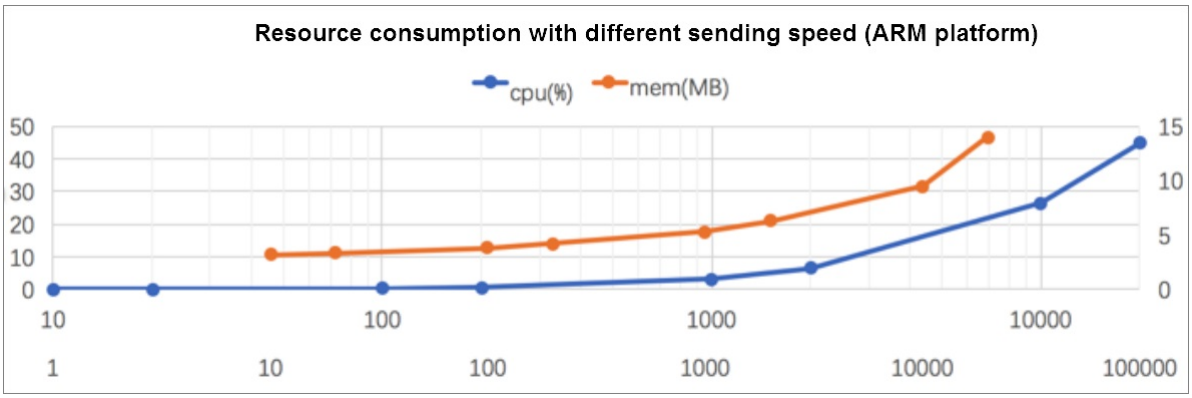
Test results

- Test results on the x86 servers
  - C Producer Library can send up to 90 MB of data per second. C Producer Library can also upload 200,000 logs per second, and occupies only 70% of CPU resources and 140 MB of memory.
  - When C Producer Library sends 200 logs per second, it occupies less than 0.01% of the CPU resources.
  - Each sending thread consumes 1.2 μs to send a log on average.



- Test results on Raspberry Pi

- The CPU of Raspberry Pi reaches a speed of only 600 MHz. Therefore, the performance of Raspberry Pi is approximately 10% of that of an X-86 server. In this case, C Producer Library can send a maximum of 20,000 logs per second.
- When Raspberry Pi sends 20 logs per second, it occupies less than 0.01% of the CPU resources.
- When Raspberry Pi is connected to a PC shared network by using a USB port, each sending thread consumes 12 μs to send a log on average.



8.2. Use web tracking to collect logs

This topic describes how to collect log data to Log Service by using the web tracking feature, and how to query and analyze the collected log data.

Context

When you send an important email, you can set the **read receipt** tag in the email. Then, you can receive a receipt when the recipient reads the email. The read receipt mode is widely used in the following scenarios:

- Check whether a recipient has read a leaflet after the leaflet is sent to the recipient.
- Check how many users have clicked a promotional web page.
- Analyze page views (PVs) of a marketing page on a mobile app.

Traditional solutions are developed for websites and webmasters. These solutions do not apply to data collection or analysis in the preceding scenarios due to the following reasons:

- Traditional solutions cannot meet personalized requirements. User behavior data is not generated at the mobile client. The user behavior data includes some parameters that are specific to personalized campaigns, such as the source, channel, environment, and behavior parameters.
- Traditional solutions are difficult and expensive to develop. To collect and analyze data, you must purchase cloud hosts, public IP addresses, servers used to receive development data, and message-oriented middleware. You must configure mutual backup to ensure the high service availability. In addition, you must develop and test the server.
- Traditional solutions are not user-friendly. After data is transmitted to the server, you must clear the data and import the data to the database. Then, you can generate data for your business operation
- Traditional solutions cannot provide auto scaling and cannot estimate the resource usage of users. Therefore, a large resource pool must be reserved.

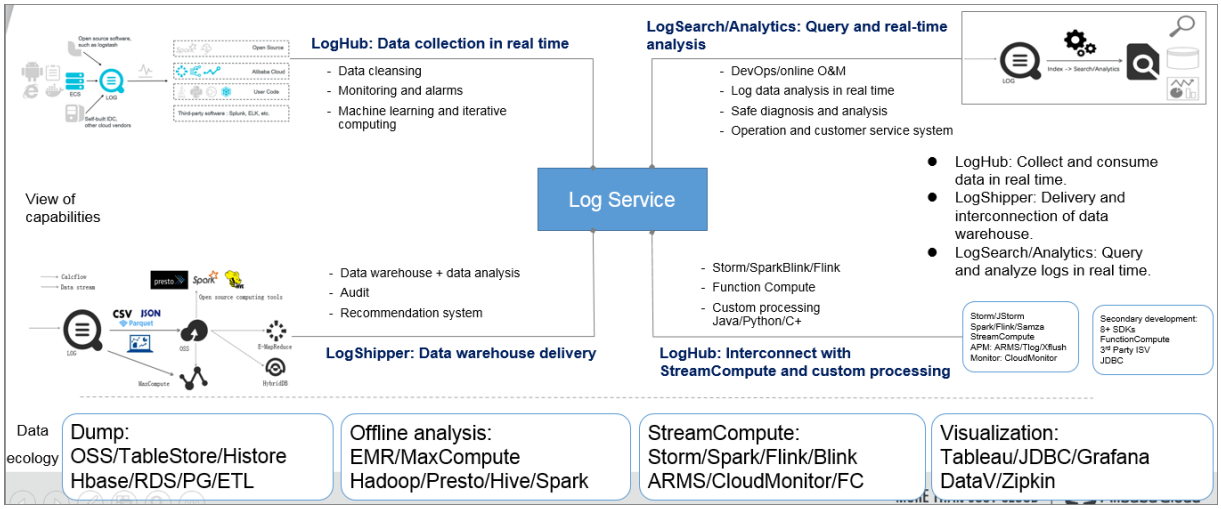
For these reasons, if you need to deliver content to intended users, these users require an efficient method to collect and analyze user behavior data.

Log Service provides Web Tracking, JavaScript, and Tracking Pixel SDKs for the preceding lightweight scenarios where you need to collect data based on tracking points. This allows you to report tracking points and data within 1 minute. In addition, Log Service provides 500 MB of free quota per month for each Alibaba Cloud account. For more information, see Pricing.

Features

Log Service is a one-stop service that is developed to collect and analyze log data. Log Service allows you to quickly collect, consume, ship, query, and analyze large amounts of log data without the need for custom development. This improves both O&M efficiency and operational efficiency. Log Service has the following modules:

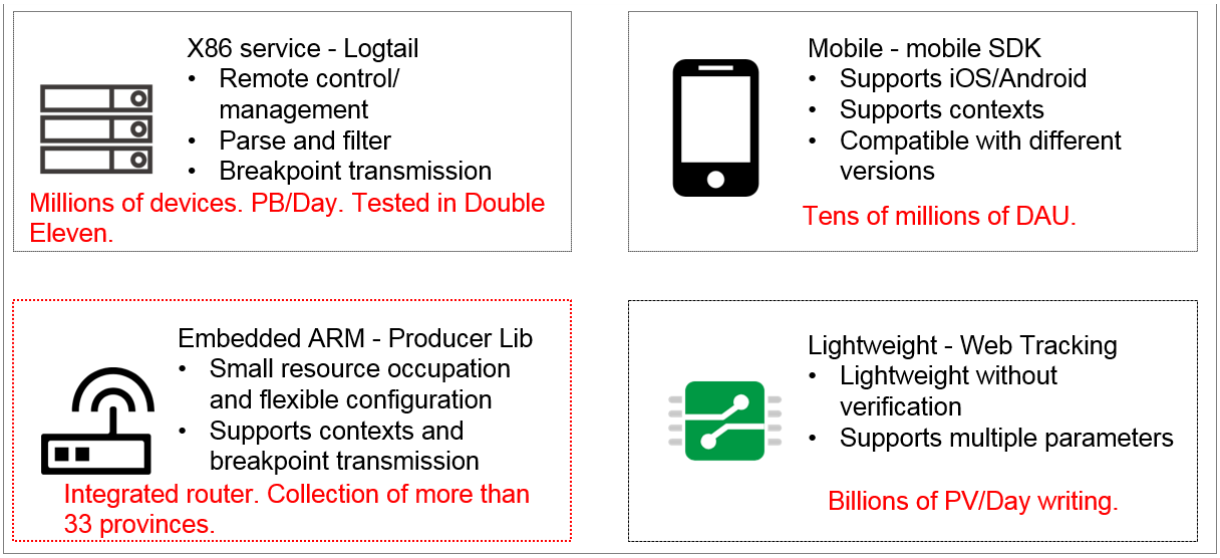
- LogHub: This module allows you to collect and consume log data in real time. LogHub is connected with Blink, Flink, Spark Streaming, Storm, and Kepler.
- LogShipper: This module allows you to ship data to consumers. LogShipper is connected with MaxCompute, E-MapReduce, Object Storage Service (OSS), and Function Compute.
- LogSearch and Analytics: This module allows you to query and analyze log data in real time. LogSearch and Analytics is connected with DataV, Grafana, Zipkin, and Tableau.



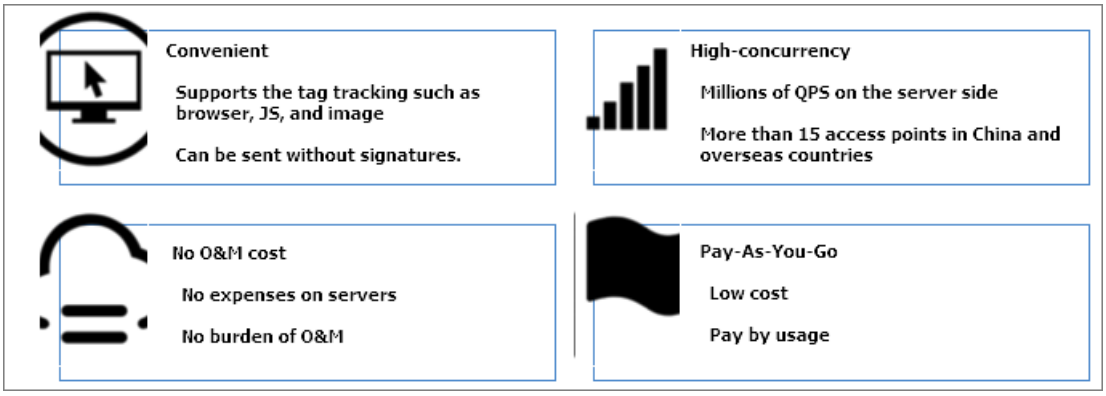
Benefits on the collection side

Log Service allows you to import data from 30 data sources and provides end-to-end solutions for servers, mobile clients, and embedded devices in multiple programming languages.

- Logtail: a log collection agent for x86 servers.
- Android or iOS SDK: an SDK for mobile clients.
- Producer Library: a library for smart devices and devices that have limited CPU or memory.



Web tracking is a lightweight collection solution in which you only need to use the HTTP GET request method to transmit data to a Log Service Logstore. This solution applies to scenarios where no verification is required to collect data from different sources, such as static web pages, online advertisements, promotional documents, and mobile clients. The following figure shows benefits of the web tracking feature.



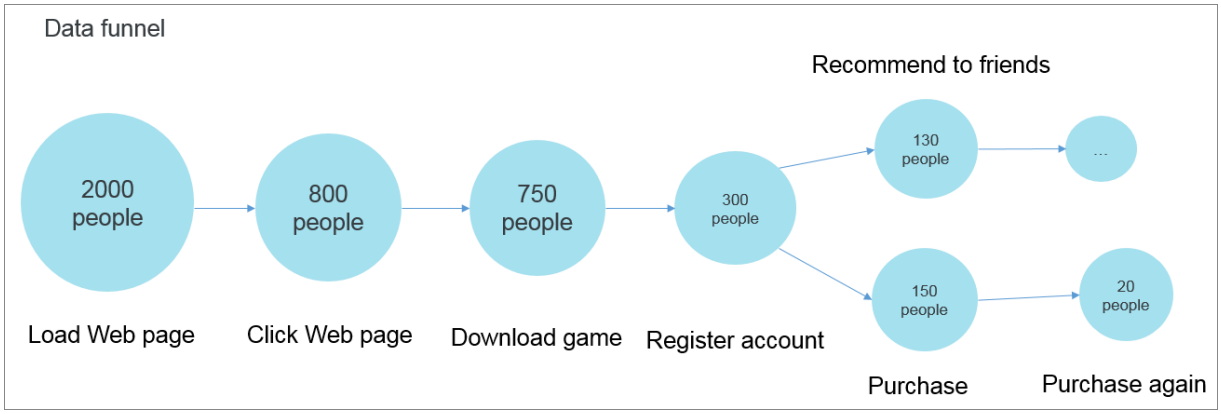
Web tracking process

Web tracking (also known as tracking pixel) is an HTML image tag. In web tracking, a 0-pixel image can be embedded on an HTML page and the image is invisible to users by default. When you access the page and the image is loaded, a GET request is initiated to transmit relevant parameters to the server. For more information, see Use web tracking to collect logs.

Scenarios

After you create new content such as new features, promotional campaigns, games, and articles, you can send the contents to users at the earliest opportunity. This is the first and most important step to attract and retain users.

For example, your company distributed 10,000 advertisements to promote a game. Only 2,000 advertisements were loaded for users, accounting for 20% of the total number of advertisements. Only 800 users clicked the adverting link. Fewer users downloaded the game, created accounts, and tried the game.



In this example, if you can monitor the effectiveness of the promotion in real time, you can promote the game with better results. To reach your promotion goals, you can use the following channels:

- Internal messages, official blogs, and homepage banners.
- SMS messages, emails, and leaflets.
- New media, such as Sina Weibo, DingTalk group, WeChat public account, Zhihu forum, and TouTiao.

Procedure

1. Enable the web tracking feature.

   Create a Logstore (for example, myclick) in Log Service and enable the web tracking feature.

2. Generate a web tracking tag.

i. Add an identifier to each promotion channel for the article (named 1001) that you want to promote. Then, generate a web tracking tag named img.

- Internal message (mailDec)

```
<img src="http://example.cn-hangzhou.log.aliyuncs.com/logstores/myclick/track_ua.gif?APIVersion=0.6.0&from=mailDec&article=1001" alt="" title="">
```

- Official website (aliyunDoc)

```
<img src="http://example.cn-hangzhou.log.aliyuncs.com/logstores/myclick/track_ua.gif?APIVersion=0.6.0&from=aliyundoc&article=1001" alt="" title="">
```

- Email (email)

```
<img src="http://example.cn-hangzhou.log.aliyuncs.com/logstores/myclick/track_ua.gif?APIVersion=0.6.0&from=email&article=1001" alt="" title="">
```

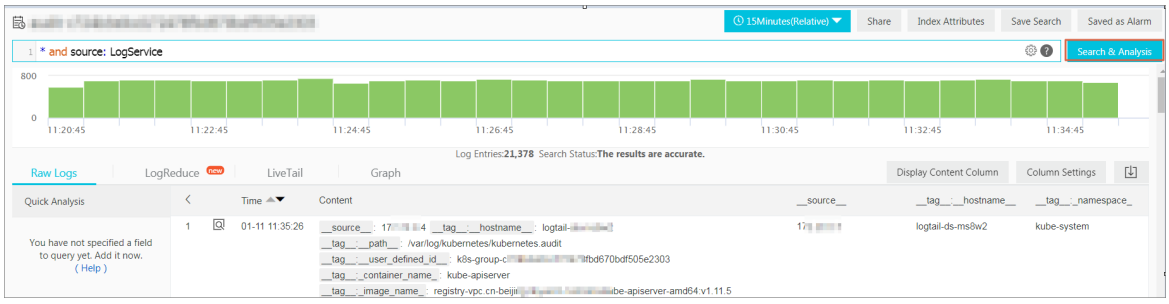
You can add more channels at the end of the "from" parameter, or add more parameters in the URL to collect their values.

ii. Add the img tag to the promotion content and release the content.

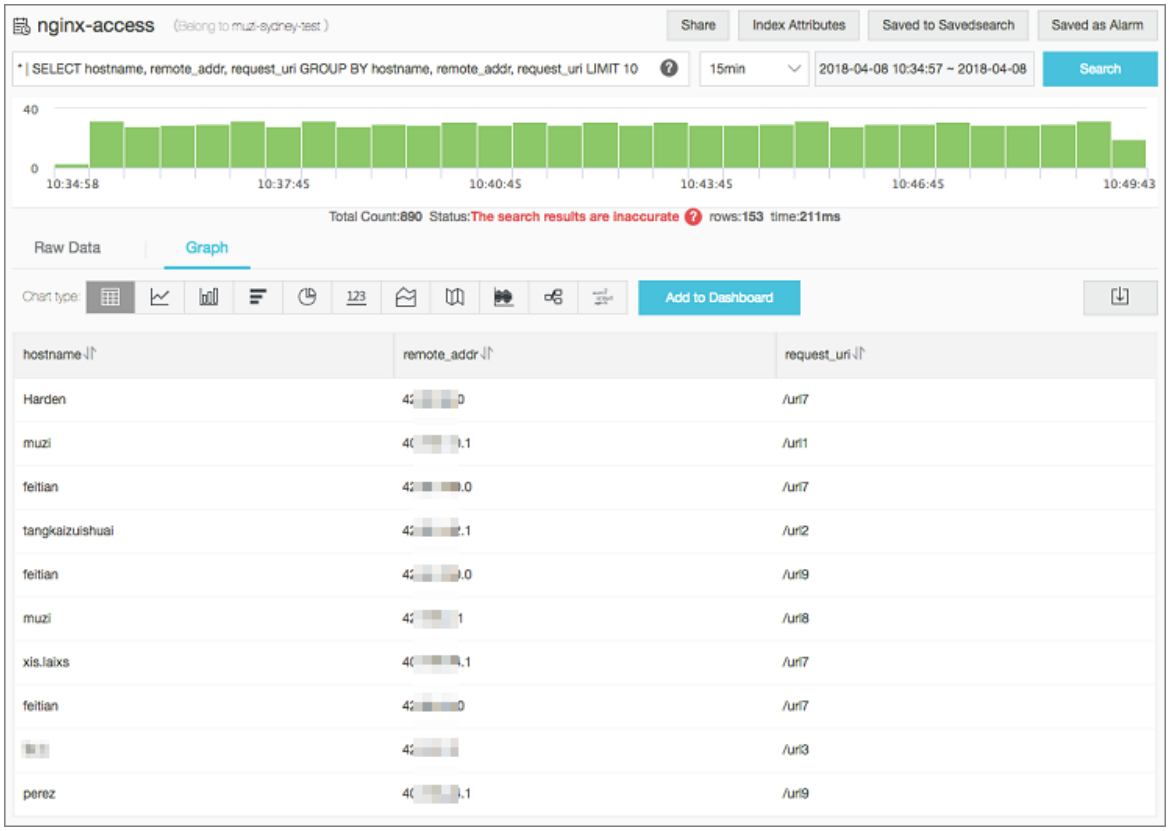
3. Analyze logs.

After you collect logs, you can use the log query and analysis feature of Log Service to query and analyze large amounts of log data in real time. For more information, see Overview. Log Service can show log analysis results on built-in dashboards and connect with DataV, Grafana, and Tableau to visualize the log analysis results. For more information, see Create a dashboard, Connect Log Service with DataV, and Connect to Log Service by using Grafana.

The following figure shows the collected log data. You can enter a keyword in the search box to query logs.



You can also enter an SQL statement after the query to analyze the log data and visualize the analysis results within seconds.



i. Create query statements.

The following examples describe how to create query statements to obtain page click and page view (PV) statistics. For more information, see Real-time log analysis.

- To query the current total traffic and PVs, execute the following statement:

```
* | select count(1) as c
```

- To query the curve of the PVs per hour, execute the following statement:

```
* | select count(1) as c, date_trunc('hour',from_unixtime(__time__)) as time grou
p by time order by time desc limit 100000
```

- To query the ratios of PVs in each channel, execute the following statement:

```
* | select count(1) as c, f group by f desc
```

- To query the devices to which the page views belong, execute the following statement:

```
* | select count_if(ua like '%Mac%')  as mac, count_if(ua like '%Windows%')  as w
in, count_if(ua like '%iPhone%')  as ios, count_if(ua like '%Android%')  as andro
id
```

- To query the locations to which the page views belong, execute the following statement:

```
* | select ip_to_province(__source__) as province, count(1) as c group by provinc
e order by c desc limit 100
```

ii. Configure a dashboard to visualize the collected data in real time.

> **Note** The img tag records your access to this topic. You can find the tag in the source code of the page.

8.3. Build a service to upload logs from mobile apps to Log Service

Mobile apps are commonly used to upload data due to the fast development of the mobile Internet. If logs can be uploaded from mobile apps to Log Service instead of being transferred by app servers, you can focus on the development of your business logic.

Context

When you write logs to Log Service in normal mode, you must use the AccessKey pair of your Alibaba Cloud account for authentication and anti-tamper protection. If a mobile app accesses Log Service in this mode, you must save your AccessKey pair on a mobile client. This increases the risk of data leaks if the AccessKey pair is exposed. If your AccessKey pair is exposed, you must upgrade the mobile app and change the AccessKey pair. This process is complex and costly. To upload logs from mobile clients to Log Service, you can also use app servers to transfer the logs. If the number of mobile apps is large, the app servers must meet high performance requirements to carry all data from mobile clients.

To prevent the preceding issues, Log Service provides a more secure and convenient solution to collect logs from mobile apps based on Resource Access Management (RAM). You can use RAM to directly transfer data. In this mode, you do not need to save your AccessKey pair on a mobile client. This prevents your AccessKey pair from being exposed. You can use a temporary token to increase data security. The temporary token has a lifecycle. You can configure access permission policies for the temporary token. For example, you can reject access requests from specified CIDR blocks.

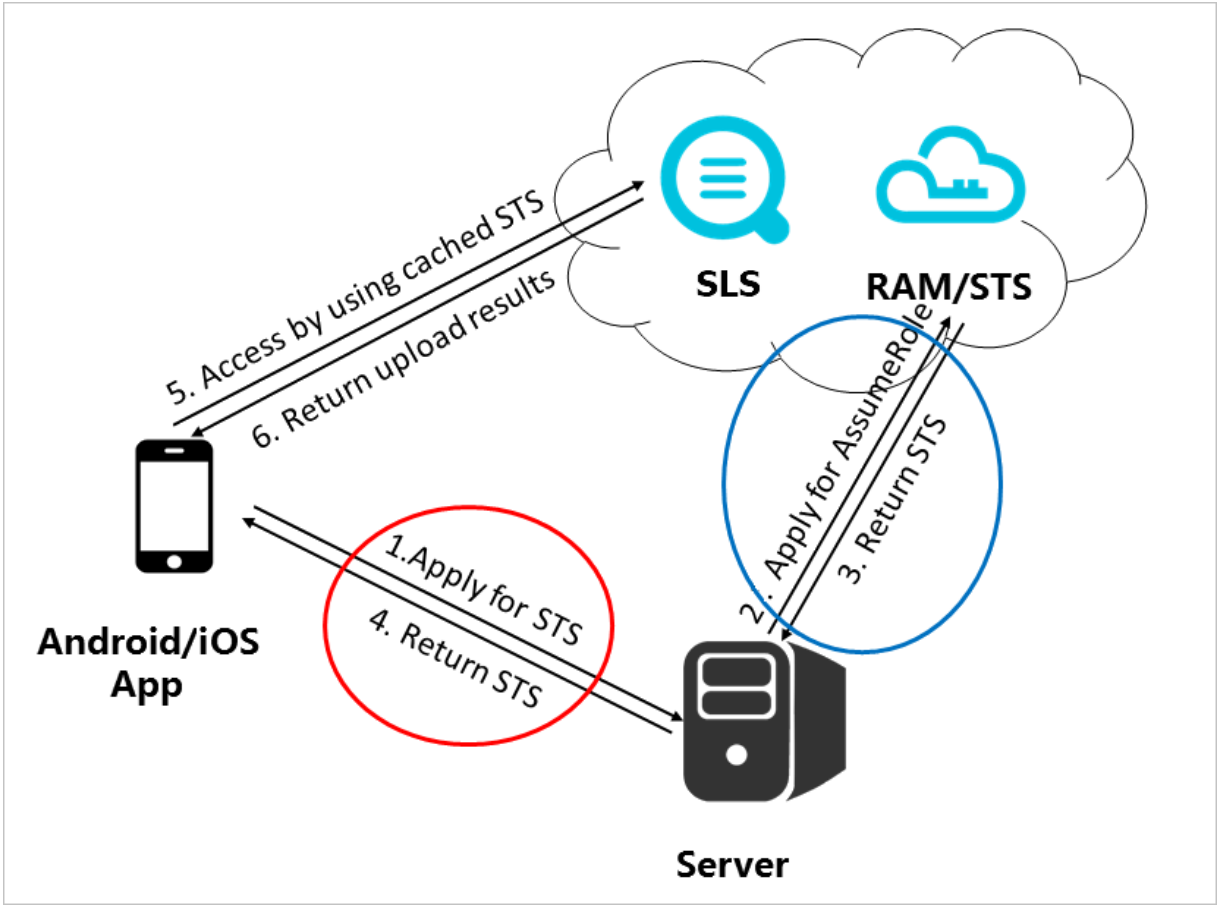
You can create a RAM role of Log Service and configure a mobile app as a RAM user to assume this role. This way, you can build a data transfer service for the mobile app based on Log Service within 30 minutes. The direct data transfer service allows mobile apps to directly access Log Service, and only the control flow is sent to app servers.

Benefits

A data transfer service that is built for mobile apps based on Log Service by using RAM has the following benefits:

- Higher access security: Flexible and temporary permission assignment and authentication are supported.
- Lower cost: Fewer servers are required. Mobile apps are directly connected to Alibaba Cloud and only the control flow is sent to app servers.
- Higher concurrency: A large number of users can use the service at the same time. Higher upload bandwidth and download bandwidth are provided by Log Service.
- Auto scaling: Log Service provides unlimited storage space.

The following figure shows the architecture.



The following table describes the nodes of the architecture.

Node	Description
Android or iOS mobile app	The app on the mobile phones of users. Logs are generated by the app.
SLS	Log Service. Log Service stores log data that is uploaded from the app.
RAM/STS	RAM. This service allows you to manage user identities and resource access permissions. You can use RAM to generate temporary upload credentials.
App server	The backend service that is developed for the Android or iOS app. The app server manages tokens that are used by the app to upload and download logs. The app server also manages the metadata that is uploaded by users to the app.

Configuration process

1. An Android or iOS app requests a temporary access credential from your app server.

To prevent data leaks, the Android or iOS app does not store the AccessKey ID or AccessKey secret. The Android or iOS app must request a temporary upload credential (a token) from your app server. The token is valid only for a specific period of time. For example, if the validity period of a token is set to 30 minutes, the Android or iOS app can use this token to access Log Service within 30 minutes. The validity period of a token can be specified by the app server. However, the app must request a new token after 30 minutes.

> **Notice** Each time the Android or iOS app obtains a token from the app server, the app server caches the token based on the validity period. We recommend that the app server sends the cached token as a response to each client request. After the cached token expires, the app server requests a new token.

2. The app server verifies the preceding request and returns a token to the Android or iOS app.
3. After the mobile app obtains the token, the mobile app can access Log Service.

This topic describes how to use an app server to request a token from RAM, and how to obtain the token for an Android or iOS app.

Procedure

1. Authorize a RAM user to manage Log Service.

   Create a RAM role and configure a mobile app as a RAM user to assume this role. For more information, see Create a RAM role whose trusted entity is an Alibaba Cloud account and authorize the RAM role to access Log Service.

   After you configure the mobile app, you can obtain the following information:

   - The AccessKey ID and AccessKey secret of the RAM user
   - The Alibaba Cloud Resource Name (ARN) of the RAM role.

2. Set up an app server.

   This topic provides sample programs in multiple languages. The download URLs are listed at the end of this topic.

   Each language pack that you download contains the configuration file config.json. The following script shows the config.json file:

   ```
   {
       "AccessKeyID" : "",
       "AccessKeySecret" : "",
       "RoleArn" : "",
       "TokenExpireTime" : "900",
       "PolicyFile": "policy/write_policy.txt"
   }
   ```

   i. AccessKeyID: the AccessKey ID of your Alibaba Cloud account. For more information, see AccessKey pair.

   ii. AccessKeySecret: the AccessKey secret of your Alibaba Cloud account.

   iii. RoleArn: the ARN of the RAM role.

   iv. TokenExpireTime: the validity period of the token that is obtained by the Android or iOS app. Valid values: 900 to 1800. Unit: seconds.

v. PolicyFile: the file that lists the permissions of the token. You can use the default value.

This topic provides the following two token files that define the permissions in the policy directory:

- write_policy.txt: grants a token the write permissions on the projects of an Alibaba Cloud account.
- readonly_policy.txt: grants a token the read permissions on the projects of an Alibaba Cloud account.

You can configure your policy file based on your business requirements.

Response format:

```
//Sample success response
{
    "StatusCode":200,
    "AccessKeyId":"STS.3p***dgagdasdg",
    "AccessKeySecret":"rpnwO9***tGdrddgsR2YrTtI",
   "SecurityToken":"CAES+wMIARKAAZhjH0EUOIhJMQBMjRywXq7MQ/cjLYg80Aho1ek0Jm63XMhr9Oc5s˙∂
˙∂3qaPer8p1YaX1NTDiCFZWFkvlHf1pQhuxfKBc+mRR9KAbHUefqH+rdjZqjTF7p2m1wJXP8S6k+G2MpHrUe6TY
BkJ43GhhTVFMuM3BZajY3VjZWOXBIODRIR1FKZjIiEjMzMzE0MjY0NzM5MTE4NjkxMSoLY2xpZGSSDgSDGAGESG
TETqOio6c2RrLWRlbW8vKgoUYWNzOm9zczoqOio6c2RrLWRlbW9KEDExNDg5MzAxMDcyNDY4MThSBTI2ODQyWg9
Bc3N1bWVkUm9sZVVzZXJgAGoSMzMzMTQyNjQ3MzkxMTg2OTExcglzZGstZGVtbzI=",
   "Expiration":"2017-11-12T07:49:09Z",
}
//Sample error response
{
    "StatusCode":500,
    "ErrorCode":"InvalidAccessKeyId.NotFound",
    "ErrorMessage":"Specified access key is not found."
}
```

The following table describes the success response parameters. The five variables in the table constitute a token.

Parameter	Description
StatusCode	The result returned when the app obtains the token. The app returns 200 if the token is obtained.
AccessKeyId	The AccessKey ID that the Android or iOS app obtains when the app initializes LogClient.
AccessKeySecret	The AccessKey secret that the Android or iOS app obtains when the app initializes LogClient.
SecurityToken	The token that the Android or iOS app uses to access Log Service.
Expiration	The expiration time of the token. The Android SDK automatically checks the validity of the token and then obtains a new token as needed.

The following table describes the error response parameters.

Parameter	Description
StatusCode	The result returned when the app obtains the token. The app returns 500 if the token fails to be obtained.
ErrorCode	The error cause.
ErrorMessage	The error description.

You can perform the following operations to run the sample code:

For Java V1.7 or later, create a Java project after you download and decompress the package. Copy the dependency, code, and configuration to the project, and then run the main function. By default, the program listens on port 7080 and waits for the HTTP request. You can perform these operations in other languages by using this method.

3. Construct an HTTP request on a mobile client to obtain a token from the app server.

   The following script shows the formats of an HTTP request and response:

```
Request URL: GET https://localhost:7080/
Response:
{
"StatusCode":"200",
"AccessKeyId":"STS.XXXXXXXXXXXXXXX",
"AccessKeySecret":"",
"SecurityToken":"",
"Expiration":"2017-11-20T08:23:15Z"
}
```

> **Note** All examples in this topic are used to demonstrate how to deploy a server. When you deploy a server, you can configure the parameters based on these examples.

Download the source code

Sample code of the app server: PHP, Java, Ruby, and Node.js.

8.4. Collect Zabbix data

Zabbix is a commonly used open source monitoring system. Zabbix provides a variety of alert rules for system monitoring. Log Service allows you to collect monitoring data from Zabbix to a Logstore. This topic describes how to collect Zabbix data to Log Service.

Prerequisites

- Zabbix is downloaded and installed. For more information, see Download and install Zabbix.

  In this topic, Zabbix is installed on an Elastic Compute Service (ECS) instance.
- A project and a Logstore are created. For more information, see Create a project and Create a Logstore.

Step 1: Specify a data storage path

Zabbix stores monitoring data on the machine on which Zabbix is installed. To specify the storage path for monitoring data, perform the following steps:

1. Log on to the ECS instance on which Zabbix is installed.
2. Open the *zabbix_server.conf* file.

   ```
   vim /etc/zabbix/zabbix_server.conf
   ```

3. Specify the data storage path in the *zabbix_server.conf* file.

   ```
   ExportDir=/tmp/
   ```

4. Restart the Zabbix service for the setting to take effect.

   ```
   systemctl restart zabbix-server
   ```

   After the setting takes effect, Zabbix generates a file whose file name extension is *.ndjson* in the */tmp* directory to store monitoring data.

Step 2: Create a Logtail configuration

1.
2. In the **Import Data** section, click **JSON - Text Log**.
3. Select the project and Logstore. Then, click **Next**.
4. Create a machine group.
   i. On the **ECS Instances** tab, select the ECS instance on which Zabbix is installed. Then, click **Execute Now**.

      For more information, see Install Logtail on ECS instances.

      If Zabbix is installed on a server in a self-managed cluster or a server on a third-party cloud, you must manually install Logtail. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.

   ii. After Logtail is installed, click **Complete Installation**.
   iii. In the **Create Machine Group** step, configure **Name** and click **Next**.

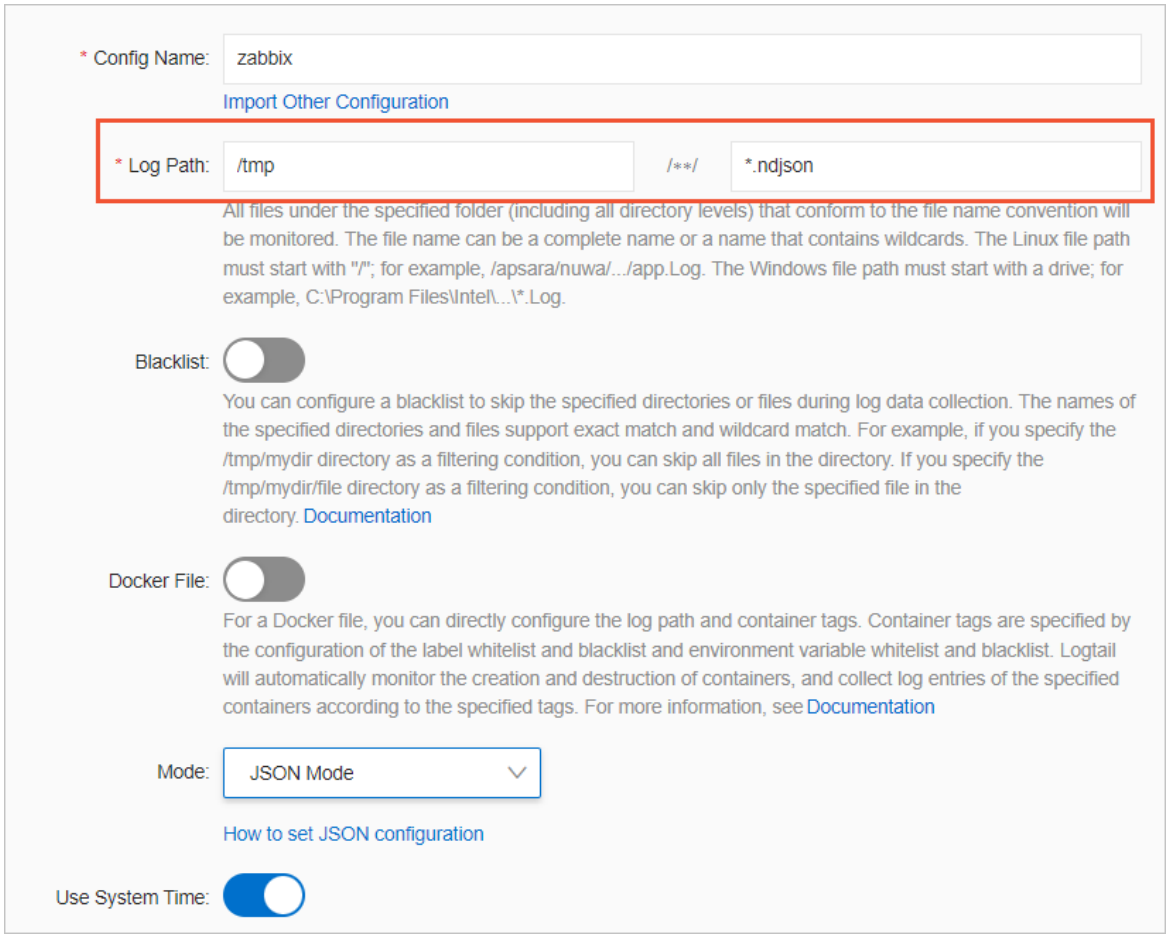
      Log Service allows you to create IP address-based machine groups and custom identifier-based machine groups. For more information, see Create an IP address-based machine group and Create a custom ID-based machine group.

5. Select and move the new machine group from **Source Server Groups** to **Applied Server Groups**. 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?

6. Create a Logtail configuration and click **Next**.

Zabbix monitoring data is of the JSON type. We recommend that you select JSON Mode for Mode in the Logtail configuration. In addition, you must set **Log Path** to the data storage path that you specify in Step Step 1: Specify a data storage path. For more information about other parameters, see Collect logs in JSON mode.



7. Preview data, configure indexes, and then click **Next**.

   By default, full-text indexing is enabled for Log Service. You can also configure field indexes based on collected logs in manual or automatic mode. For more information, see Configure indexes.

   > **Note** If you want to query and analyze logs, you must enable full-text indexing or field indexing. If you enable both full-text indexing and field indexing, the system uses only field indexes.

8.5. Collect logs across Alibaba Cloud accounts

This topic describes how to collect logs from a server across Alibaba Cloud accounts.

Context

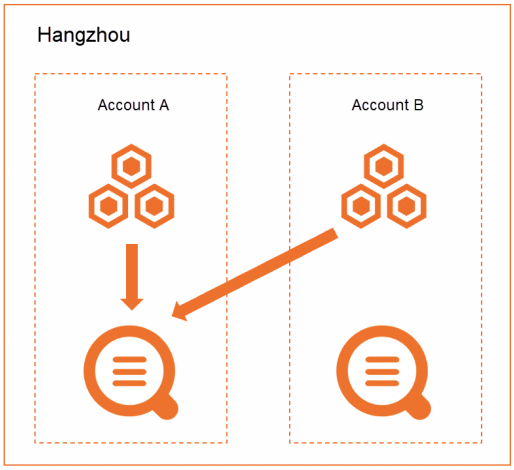
If you want to use Logtail to collect logs from a server, you must install Logtail on the server, and then configure the ID of the Alibaba Cloud account for which Log Service is activated as a user identifier on the server. This way, the Alibaba Cloud account can use Logtail to collect logs from the server. If you do

not configure a user identifier on a server, Log Service cannot receive the heartbeat status of the server and Logtail cannot collect logs from the server.

For example, an e-commerce enterprise has two e-commerce applications that are deployed on Elastic Compute Service (ECS) clusters in the China (Hangzhou) region. The enterprise uses two Log Service projects that reside in the China (Hangzhou) region to manage logs.

- Application A is deployed on a Linux ECS cluster that belongs to Alibaba Cloud Account A (12****456) and Log Service is activated for the account to manage logs.
- Application B is deployed on a Linux ECS cluster that belongs to Alibaba Cloud Account B (17****397) and Log Service is activated for the account to manage logs.

The enterprise wants to use Log Service that is activated for Alibaba Cloud Account A (12****456) to collect the logs of the two applications and store the logs in two Logstores of the same project. In this case, you must create a Logtail configuration, a machine group, and a Logstore to collect and store the logs of Application B. The Logtail configuration, machine group, and Logstore that are configured for Application A remain unchanged.



Step 1: Create a user identifier file

1. Log on to an ECS instance that belongs to Alibaba Cloud Account B.

   > **Notice** You must create a user identifier file on each ECS instance of ECS Cluster B.

2. Run the following command to create a user identifier file.

   In this example, a file named the ID of Alibaba Cloud Account A is created. For more information, see Configure a user identifier.

   ```
   touch /etc/ilogtail/users/12****456
   ```

Step 2: Create a custom ID-based machine group

1. Create a custom ID file for the machine group on an ECS instance.

   > **Notice** You must create a custom ID file for the machine group on each ECS instance of ECS Cluster B.

   i. Log on to an ECS instance that belongs to Alibaba Cloud Account B.

ii. Create a file named user_defined_id in the /etc/ilogtail/ directory and specify a custom ID in the */etc/ilogtail/user_defined_id* file.

For example, if you want to set the custom ID to `application_b`, specify `application_b` in the file, and then save the file. For information about file paths, see Create a custom ID-based machine group.

2. Create a machine group in the Log Service console.

i. Use Alibaba Cloud Account A to log on to the Log Service console.

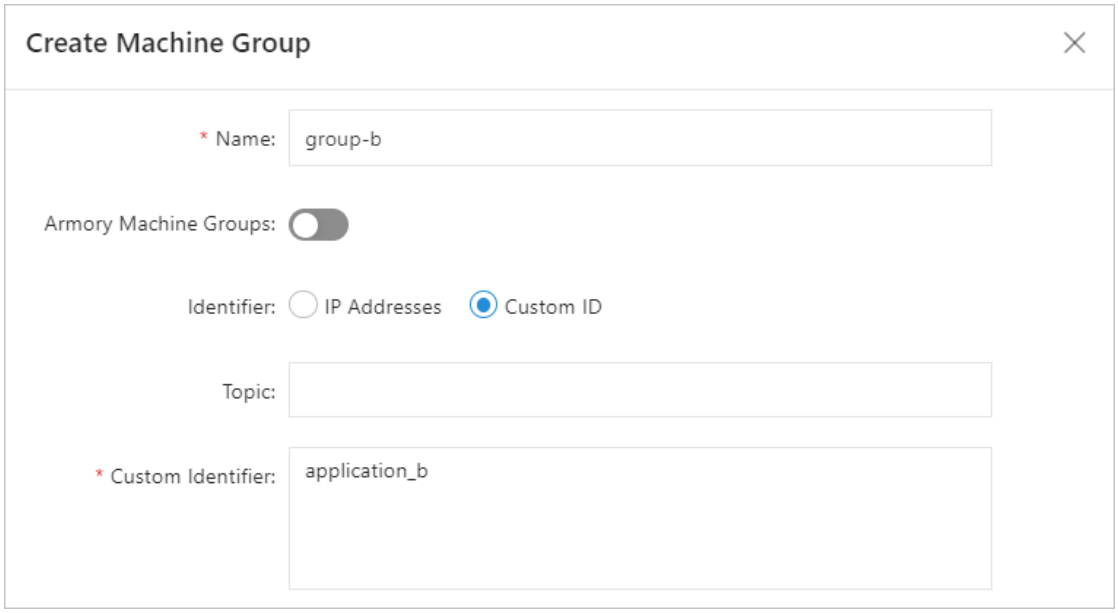
ii. In the Projects section, click the project that you want to manage.

iii. In the left-side navigation pane, choose **Resources > Machine Groups**.

iv. On the **Machine Groups** tab, choose > **Create Machine Group**.

v. In the **Create Machine Group** panel, set the parameters and click **OK**, as shown in the following figure.

In the **Custom Identifier** field, enter the custom ID that you specified in Step . For information about other parameters, see Create a custom ID-based machine group.

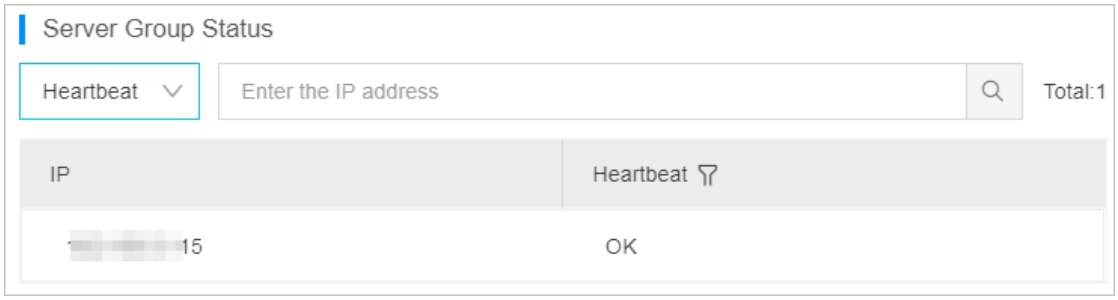


3. Check whether the heartbeat status of each server in the machine group is OK.

i. In the Machine Groups list, click the machine group that you created.

ii. On the **Machine Group Settings** page, view the status of the machine group. You can view the list of servers that use the same custom ID. You can also view the heartbeat status of each server.

If the **Heartbeat** status is **OK**, the ECS instance is connected to Log Service. If the status is **FAIL**, see What can I do if the Logtail client has no heartbeat?

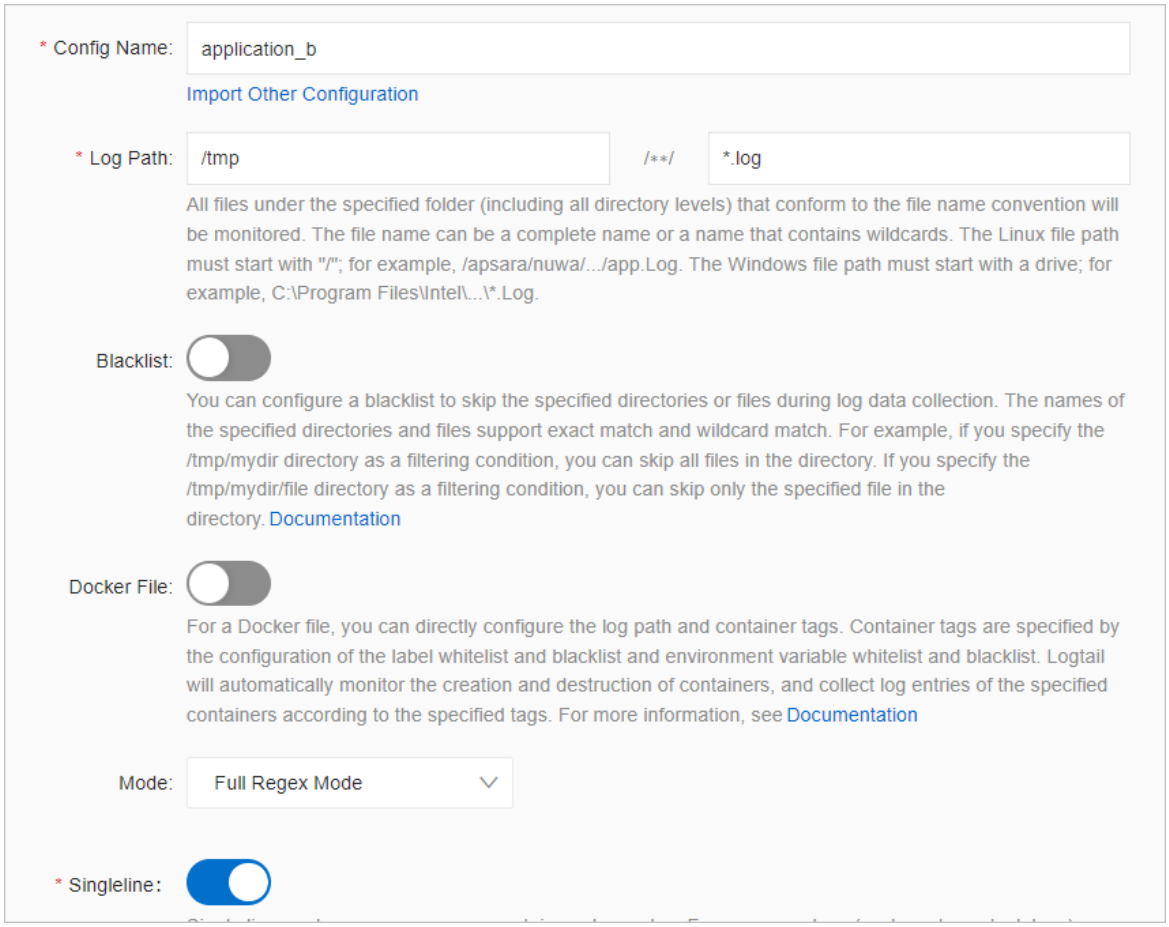


Step 3: Collect logs

1. Use Alibaba Cloud Account A to log on to the Log Service console.
2. In the **Import Data** section, select **RegEx - Text Log**.
3. In the **Specify Logstore** step, select the project and the Logstore, and 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 created in Step 2, move the machine group from the **Source Server Groups** section to the **Applied Server Groups**, and then click **Next**.
6. Create a Logtail configuration and click **Next**.

   For information about the parameters, see Collect logs in full regex mode.

> **Notice**
>
> - By default, you can use only one Logtail configuration to collect each log file. The collection process of Logtail in Alibaba Cloud Account B is not stopped. In this case, the Logtail configuration of Alibaba Cloud Account A cannot take effect. To make sure that the Logtail configuration of Alibaba Cloud Account A takes effect, you can use one of the following methods:
>   - Stop the collection process in Alibaba Cloud Account B. To stop the collection process, use Alibaba Cloud Account B to log on to the Log Service console and remove the original Logtail configuration from the machine group. For more information, see Manage Logtail configurations.
>   - Add compulsory collection settings to the Logtail configuration of Alibaba Cloud Account A. For more information, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?.
> - After you create the Logtail configuration, delete the original Logtail configuration of Alibaba Cloud Account B to prevent repeated collection of logs. For more information, see Delete Logtail configurations.



7. Preview data, configure indexes, and then click **Next**.

   By default, Log Service enables full-text indexing. You can configure field indexes based on the logs that are collected in manual mode or automatic mode. For more information, see Configure indexes.

Related operations

If you want to migrate historical data from Alibaba Cloud Account B to the current Logstore, you can create a data transformation task in the original Logstore, and then replicate the data to the current Logstore. For more information, see Replicate data from a Logstore.

> **Notice**
>
> If you create a data transformation task to transform data across Alibaba Cloud accounts, you must use a custom role or an AccessKey pair to grant the required permissions for the task. In this example, a custom role is used.
>
> - The first **role ARN** is used to grant the custom role or AccessKey pair the required permissions to read data from a source Logstore. For information about how to grant the required permissions to a RAM role, see Grant the RAM role the permissions to read data from a source Logstore.
> - The second **role ARN** is used to grant the custom role or AccessKey pair the required permissions to write transformation results to a destination Logstore. For information about how to grant the required permissions to a RAM role, see Grant the RAM role the permissions to write data to destination Logstores across Alibaba Cloud accounts.

8.6. Collect container logs across Alibaba Cloud accounts

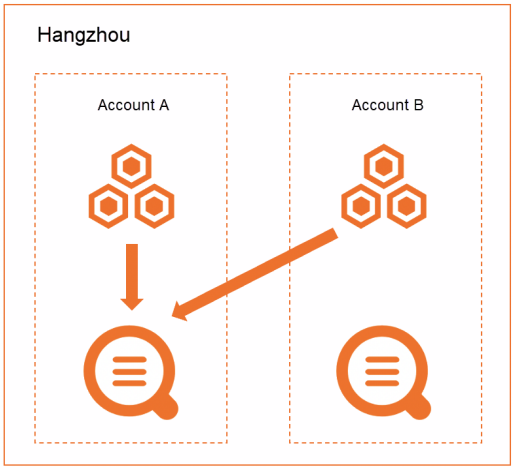
This topic describes how to collect container logs from Container Service for Kubernetes (ACK) across Alibaba Cloud accounts.

Context

For example, an e-commerce enterprise has two e-commerce applications that are deployed on ACK clusters in the China (Hangzhou) region. The enterprise uses two Log Service projects that reside in the China (Hangzhou) region to manage logs.

- Application A is deployed on an ACK cluster that belongs to Alibaba Cloud Account A (12****456) and Log Service is activated for the account to manage logs.
- Application B is deployed on an ACK cluster that belongs to Alibaba Cloud Account B (17****397) and Log Service is activated for the account to manage logs.

The enterprise wants to use Log Service that is activated for Alibaba Cloud Account A (12****456) to collect the logs of the two applications and store the logs in two Logstores of the same project. In this case, you must create a Logtail configuration, a machine group, and a Logstore to collect and store the logs of Application B. The Logtail configuration, machine group, and Logstore that are configured for Application A remain unchanged.



Step 1: Configure the ID of an Alibaba Cloud account as a user identifier

1. Use Alibaba Cloud Account B to log on to the ACK console.
2. Configure the ID of Alibaba Cloud Account A as a user identifier.
   i. In the left-side navigation pane, click **Clusters**.
   ii. On the **Clusters** page, click the cluster that you want to manage.
   iii. In the left-side navigation pane, choose **Configurations > ConfigMaps**.
   iv. Set the **Namespace** parameter to **kube-system**. In the ConfigMap list, find **alibaba-log-configuration** and click **Edit** in the Actions column.
   v. In the **Edit** panel, configure the following configuration and click **OK**.

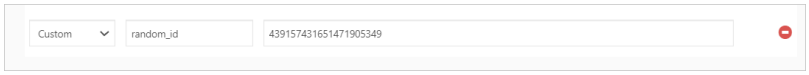
      Add the ID of Alibaba Cloud Account A to the **log-ali-uid** file, and then obtain the value of the **log-machine-group** parameter, for example, k8s-group-cc47****54428. When you create a machine group, specify the value for the **Custom Identifier** parameter.

      Separate multiple account IDs with commas (,). Example: `17****397,12****456`.

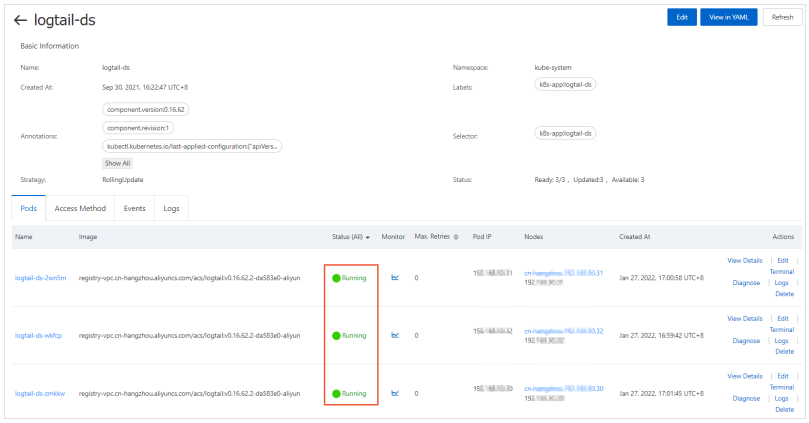
3. Restart logtail-ds for the settings to take effect.
   i. In the left-side navigation pane, choose **Workloads > DaemonSets**.
   ii. In the DaemonSets list, find **logtail-ds** and click **Edit** in the Actions column.
   iii. In the **Environment Variable** section, click **Add**.

iv. Add a custom variable and specify an arbitrary key-value pair, for example, random_id: 439157431651471905349.



v. Click **Update**.

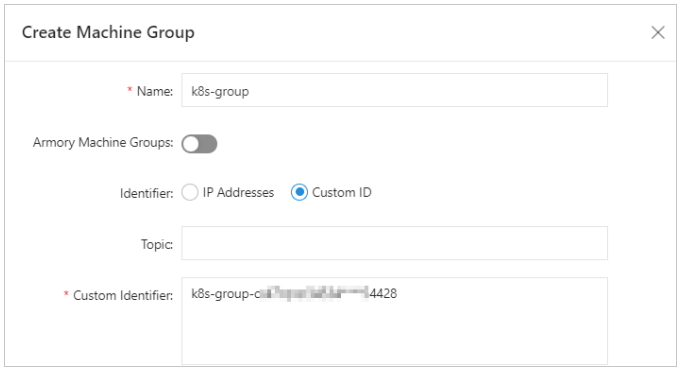
On the details page of **logtail-ds**, check whether each container pod is in the **Running** state and whether the time when each pod is created is the same as the time when you update the settings.



Step 2: Create a machine group

1. Use Alibaba Cloud Account A to log on to the Log Service console.
2. In the Projects section, click the project that you want to manage.
3. In the left-side navigation pane, choose **Resources > Machine Groups**.
4. On the **Machine Groups** tab, choose > **Create Machine Group**.
5. In the **Create Machine Group** panel, set the parameters and click **OK**, as shown in the following figure.

   In the **Custom Identifier** field, enter the machine group identifier that you obtained in Step 1: Configure the ID of an Alibaba Cloud account as a user identifier, for example, k8s-group-cc47****54428. For information about other parameters, see Create a custom ID-based machine group.



6. Check whether the heartbeat status of each server in the machine group is OK.
   i. In the Machine Groups list, click the machine group that you created.

ii. On the **Machine Group Settings** page, view the status of each Elastic Compute Service (ECS) instance.

If the **Heartbeat** status is **OK**, the ECS instance is connected to Log Service. If the status is **FAIL**, see What can I do if the Logtail client has no heartbeat?

IP	Heartbeat
19[redacted]30	OK
17[redacted]1	OK
19[redacted]32	OK
19[redacted]31	OK

Step 3: Create a Logtail configuration

1. Use Alibaba Cloud Account A to log on to the Log Service console.
2. In the **Import Data** section, click **Kubernetes - Object**.
3. Select a project and a Logstore. Then, click **Next**.
4. Click **Use Existing Machine Groups**.
5. Select the machine group that you created in Step 2: Create a machine group, move the machine group from the **Source Server Groups** section to the **Applied Server Groups**, and then click **Next**.
6. Set the parameters for the Logtail configuration and click **Next**.

   For information about the parameters, see Use the Log Service console to collect container text logs in DaemonSet mode.

   > **Notice**
   > - By default, you can use only one Logtail configuration to collect each log file. The collection process of Logtail in Alibaba Cloud Account B is not stopped. In this case, the Logtail configuration of Alibaba Cloud Account A cannot take effect. To make sure that the Logtail configuration of Alibaba Cloud Account A takes effect, you can use one of the following methods:
   >   - Stop the collection process in Alibaba Cloud Account B. To stop the collection process, use Alibaba Cloud Account B to log on to the Log Service console and remove the original Logtail configuration from the machine group. For more information, see Manage Logtail configurations.
   >   - Add compulsory collection settings to the Logtail configuration of Alibaba Cloud Account A. For more information, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?.
   > - After you create the Logtail configuration, delete the original Logtail configuration of Alibaba Cloud Account B to prevent repeated collection of logs. For more information, see Delete Logtail configurations.

7. Preview data, configure indexes, and then click **Next**.

   By default, Log Service enables full-text indexing. You can configure field indexes based on the logs that are collected in manual mode or automatic mode. For more information, see Configure indexes.

Related operations

If you want to migrate historical data from Alibaba Cloud Account B to the current Logstore, you can create a data transformation task in the original Logstore, and then replicate the data to the current Logstore. For more information, see Replicate data from a Logstore.

> **Notice**
>
> If you create a data transformation task to transform data across Alibaba Cloud accounts, you must use a custom role or an AccessKey pair to grant the required permissions for the task. In this example, a custom role is used.
>
> - The first **role ARN** is used to grant the custom role or AccessKey pair the required permissions to read data from a source Logstore. For information about how to grant the required permissions to a RAM role, see Grant the RAM role the permissions to read data from a source Logstore.
> - The second **role ARN** is used to grant the custom role or AccessKey pair the required permissions to write transformation results to a destination Logstore. For information about how to grant the required permissions to a RAM role, see Grant the RAM role the permissions to write data to destination Logstores across Alibaba Cloud accounts.

9.FAQ

9.1. FAQ about data collection

This topic lists some frequently asked questions about the data collection feature of Log Service.

- FAQ about Logtail
- What do I do if errors occur when I use Logtail to collect logs?
- How do I use the Logtail automatic diagnostic tool?
- What do I do if Log Service detects no heartbeat connections from Logtail?
- Do I need to update Logtail configurations after the network type is changed from classic network to VPC?
- What do I do if the IP address is empty in the app_info.json file of Logtail?
- How do I collect logs from servers in a corporate intranet?
- Troubleshoot log collection exceptions in containers
- Query local collection status
- How do I view Logtail collection errors?
- How do I troubleshoot the common errors that occur when Log Service collects logs?
- How do I debug regular expressions?
- How do I optimize regular expressions?
- How do I collect different types of logs when Full Regex Mode is used?
- How do I collect logs from containers in Kubernetes clusters?
- Why am I unable to collect SLB access logs?
- What are the differences among log collection agents?
- What are the differences between LogHub and Kafka?
- What do I do if error messages appear after I install Logtail on a Windows ECS instance?

9.2. Log management

How does Log Service store and manage user logs?

A Logstore is the basic unit for storing and querying logs in Log Service. It is generally used to store a type of log data. Currently, you can add, delete, modify, and query Logstores in the Log Service console or by using the API. After a Logstore is created, you can write logs to the Logstore by using the API or SDK. Log Service also provides Logtail to help you easily collect log data from Alibaba Cloud Elastic Compute Service (ECS) instances.

Are logs lost when I delete a Logstore?

If you delete a Logstore, you also delete all the log data stored in the Logstore. Therefore, exercise caution when performing this operation.

How long is log data stored in Log Service? Can I change the data retention period?

The following features of Log Service involve the data retention period of logs:

- LogHub and LogSearch: You can set the data retention period as required.
- LogShipper: After logs are shipped to Object Storage Service (OSS) or MaxCompute, you can set the data retention period in OSS or MaxCompute.

How can I reduce the expenditure on Log Service if I want to store logs to OSS?

Log Service charges you certain fees when you use its powerful indexing and analysis features. If you only want to store logs to OSS and do not have any custom requirements for log query and analysis, you can use the following methods to reduce your expenditure on Log Service:

Important notes

- The indexing feature is disabled by default. If the indexing and analysis features are disabled, you can shorten the data retention period of a Logstore to reduce data storage costs.
- After you disable the indexing and analysis features, relevant features also become unavailable, such as log query by keyword, log statistical analysis, dashboards, and alerting. Therefore, exercise caution when performing this operation.
- Change the data retention period of a Logstore.

  You can change the data retention period of a Logstore to one day. Log Service charges certain fees for storing data in a Logstore. Therefore, you can shorten the data retention period to reduce the expenditure.
- Disable the indexing feature.
  i. Enable the OSS data shipping feature to ship log data from a Logstore to OSS for storage in a quasi-real-time manner.
  ii. On the **Logstores** tab in the left-side navigation pane, click **Search & Analysis** next to a Logstore.

iii. Delete all indexes and disable the indexing feature.

After the indexing and analysis features are disabled, Log Service charges you only small fees for using LogHub. For more information, see Pay-as-you-go.

9.3. FAQ about Logtail

What is Logtail?

Logtail is a log collection agent that is provided by Log Service. Logtail allows you to collect logs from different data sources to Log Service. After you install Logtail on the server from which you want to collect logs, Logtail monitors the specified log files and uploads logs that are newly written to the log files to a specified Logstore.

Can Logtail collect data from static log files?

Logtail monitors modification events in the file system to determine whether log files are modified. If log files are modified, Logtail collects the logs that are generated in real time and sends the logs to Log Service. If log files are not modified, Logtail does not collect data from the log files.

Which operating systems does Logtail support?

- Linux

  Logtail supports the following versions of 64-bit Linux x86 servers:

  - Alibaba Cloud Linux 2.1903
  - Red Hat Enterprise Linux 6, Red Hat Enterprise Linux 7, and Red Hat Enterprise Linux 8
  - CentOS Linux 6, CentOS Linux 7, and CentOS Linux 8
  - Debian GNU/Linux 8, Debian GNU/Linux 9, and Debian GNU/Linux 10
  - Ubuntu 14.04, Ubuntu 16.04, Ubuntu 18.04, and Ubuntu 20.04
  - SUSE Linux Enterprise Server 11, SUSE Linux Enterprise Server 12, and SUSE Linux Enterprise Server 15
  - openSUSE Leap 15.1, openSUSE Leap 15.2, and openSUSE Leap 42.3
  - Linux operating systems based on Glibc 2.5 or later versions

- Windows

  Logtail supports Microsoft Windows Server 2008 and Microsoft Windows 7 regardless of whether they use x86 or x86_64. Logtail supports other versions of Windows operating systems only if the operating systems use x86_64.

  - Microsoft Windows Server 2008
  - Microsoft Windows Server 2012
  - Microsoft Windows Server 2016
  - Microsoft Windows Server 2019
  - Microsoft Windows 7
  - Microsoft Windows 10
  - Microsoft Windows Server Version 1909
  - Microsoft Windows Server Version 2004

How do I install and upgrade Logtail?

- For more information about how to install Logtail, see Install Logtail on ECS instances, Install Logtail on a Linux server, or Install Logtail on a Windows server.
- For more information about how to upgrade Logtail, see Upgrade Logtail on a Linux server or Upgrade Logtail on a Windows server.

> **Note** If Logtail is running, you must manually upgrade Logtail.

How do I configure Logtail to collect logs?

Log Service allows you to collect text logs and container logs by using Logtail. You can also collect logs by using Logtail plug-ins. For more information, see the following topics:

- Collect text logs
- Collect container logs
- Collect logs by using Logtail plug-ins

How does Logtail collect logs?

Logtail collects logs in the following process: monitors log files, reads log files, processes logs, filters logs, aggregates logs, and sends logs. For more information, see Log collection process of Logtail.

Does Logtail support log file rotation?

Yes, Logtail supports log file rotation. For example, during log file rotation for the app.LOG file, the app.LOG.1 and app.LOG.2 files are generated. Logtail can detect the process of log file rotation and ensure that no logs are lost during this process.

How does Logtail handle network exceptions?

If a network error occurs or the write quota is exhausted, Logtail stops reading the logs that are being collected, keeps the log files open, and retries later.

What is the collection latency when Logtail collects logs?

Logtail collects logs based on the monitoring of modification events and sends the collected logs to Log Service within three seconds.

How do I collect historical logs?

If the interval from the time when a log is generated to the system time when Logtail processes the log exceeds 5 minutes, the log is considered a historical log. By default, Logtail collects only incremental logs. If you want to collect historical logs, you can use the historical log import feature that is provided by Logtail. For more information, see Import historical logs.

When does a Logtail configuration take effect after I modify it?

After you modify a Logtail configuration in the Log Service console, the Logtail configuration takes effect within three minutes.

How do I resolve issues that occur when Logtail collects logs?

You can use the following methods to resolve log collection issues. For more information, see What do I do if errors occur when I use Logtail to collect logs?.

1. Check whether the heartbeat status of Logtail is OK.
2. Check whether the logs in the specified log files are generated in real time.
3. Check whether the regular expression in the Logtail configuration matches the content of the logs.

9.4. How do I collect logs from servers in a corporate intranet?

This topic uses NGINX as an example to describe how to collect logs from servers in a corporate intranet to Log Service.

Prerequisites

A project and a Logstore are created. For more information, see Create a project and Create a Logstore.

Context

For example, you deployed multiple servers in a corporate intranet, and the servers do not have access to the Internet. If you want to collect logs from the servers to Log Service for query and analysis, you can authorize one of the servers to access the Internet. Then, you can configure this server as a gateway server. This way, you can collect logs from the other servers to Log Service.

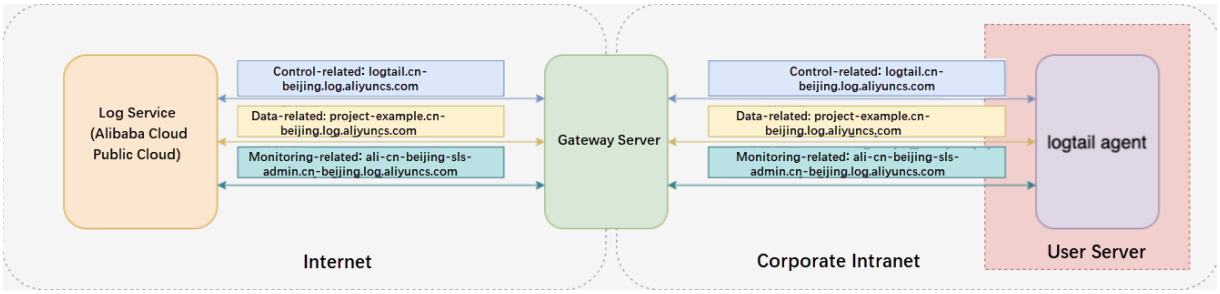
You can configure a reverse proxy server, such as NGINX, as the gateway server. NGINX is an open source and high-performance HTTP server and reverse proxy server. For more information, visit the official NGINX website.

Working principle

You can use a gateway server to collect logs from servers in a corporate intranet to Log Service. The following three types of endpoints are used when Logtail communicates with Log Service:

- The endpoints that start with `logtail`. Format: `logtail.${region}.log.aliyuncs.com`. Example: `logtail.cn-beijing.log.aliyuncs.com`. This type of endpoint is used for communication that involves control-related requests.
- The endpoints that start with a project name. Format: `${project_name}.${region}.log.aliyuncs.com`. Example: `project_example.cn-beijing.log.aliyuncs.com`. This type of endpoint is used for communication that involves data-related requests.
- The endpoints that start with `ali-${region}-sls-admin`. Format: `ali-${region}-sls-admin.${region}.log.aliyuncs.com`. Example: `ali-cn-beijing-sls-admin.cn-beijing.log.aliyuncs.com`. This type of endpoint is used to report monitoring data.

In the preceding formats, `${region}` specifies the region of the project that is used, and `${project_name}` specifies the name of the project.



Step 1: Enable anonymous write

Submit a ticket to enable anonymous write.

Step 2: Configure a gateway server

The following procedure describes how to use NGINX to configure a server that has access to the Internet in a corporate intranet as a gateway server:

1. Log on to the server that you want to configure as a gateway server.

2. Install NGINX.

   For more information, see Install NGINX.

3. Add the following settings to the *nginx.conf* file.

   By default, HTTP access is used. The following example uses the default settings. You must replace *${DNS server address}* with the actual value.

```
server {
     listen       80;
     server_name  *.log.aliyuncs.com;
     location / {
         resolver ${DNS server address};
         proxy_set_header  Host $host;
         proxy_set_header X-Forwarded-For $remote_addr;
         proxy_pass http://$host:80$request_uri;
         break;
     }
 }
```

Step 3: Bind the gateway server to the servers in the corporate intranet

After you configure the gateway server, you must bind the gateway server to the servers in the corporate intranet.

1. Log on to a server in the corporate intranet.
2. Install Logtail.
   - To install Logtail on a Linux server, follow the instructions that are provided in Install Logtail on a Linux server.
   - To install Logtail on a Windows server, follow the instructions that are provided in Install Logtail on a Windows server.
3. Configure DNS records.

   In this example, dnsmasq and Linux are used.

   i. Add the following script to the */etc/resolv.conf* file to configure the local server as the DNS server:

```
nameserver 127.0.0.1
```

   ii. Add the following script to the */etc/dnsmasq.conf* file to bind the gateway server to the local server:

   Replace *${IP address of the gateway server}* with the actual value.

```
address=/.log.aliyuncs.com/${IP address of the gateway server}
```

4. Repeat Steps 1 to to bind the gateway server to the other servers in the corporate intranet.

Step 4: Test network connectivity

1. Log on to a server in the corporate intranet.
2. Run the following commands.

In the following commands, *${region}* specifies the region of the project that is used, and *${project_name}* specifies the name of the project. Replace the variables with the actual values.

```
curl http://logtail.${region}.log.aliyuncs.com
```

```
curl http://${project_name}.${region}.log.aliyuncs.com
```

```
curl http://ali-${region}-sls-admin.${region}.log.aliyuncs.com
```

If information similar to the following code is returned, the network is connected:

```
null
```

3. Repeat Steps to to test the network connectivity for the other servers in the corporate intranet.

FAQ

If issues occur during collection, you can submit a ticket to contact technical support.

9.5. Troubleshoot log collection exceptions in containers

This topic provides solutions to exceptions that may occur when you use a Logtail container (a common container or Kubernetes) to collect logs.

Troubleshooting operations:

- Troubleshoot heartbeat exceptions in a machine group
- Troubleshoot log collection exceptions in a container

Other O&M operations:

- Log on to the Logtail container
- View Logtail operational logs
- View Logtail standard output (stdout)
- View the status of log-related components in a Kubernetes cluster
- View the version information, IP address, and time of Logtail
- What do I do if I mistakenly delete a Logstore that is created through CRD?

Troubleshoot heartbeat exceptions in a machine group

You can determine whether the Logtail on a container is correctly installed by checking the heartbeat status of a machine group.

1. Check the heartbeat status of the machine group.
   i.
   ii. In the left-side navigation pane, click **Logtail Machine Group**.
   iii. Find the target machine group and click **Status**.

   Record the number of nodes for which heartbeat status is **OK**.
2. Check the number of Worker nodes in the cluster.

Run `kubectl get node | grep -v master` to view the number of Worker nodes.

```
$kubectl get node | grep -v master
NAME                                STATUS    ROLES     AGE       VERSION
cn-hangzhou.i-bp17enxc2us3624wexh2  Ready     <none>    238d      v1.10.4
cn-hangzhou.i-bp1ad2b02jtqd1shi2ut  Ready     <none>    220d      v1.10.4
```

3. Compare whether the number of the nodes with heartbeat status of **OK** is the same as the number of Worker nodes. Then, use an appropriate troubleshooting method according to the following possible comparison results:
   - The heartbeat status of all nodes is **Failed**.
     - If you use standard Docker logs, check whether *${your_region_name}*, *${your_aliyun_user_id}*, and *${your_machine_group_user_defined_id}* are correct by following the instructions provided in parameter description.
     - If you use installation for Kubernetes on Alibaba Cloud Container Service, open a ticket.
     - If you use self-built Kubernetes installation, check whether *{your-project-suffix}*, *{regionId}*, *{ali uid}*, *{access-key-id}*, and *{access-key-secret}* are correct by following the instructions provided in parameter description. If the parameters are incorrect, run `helm del --purge alibaba-log-controller` to delete the installation package and reinstall Kubernetes.
   - The number of nodes for which the heartbeat status is OK is smaller than the number of Worker nodes.
     a. Determine whether to use the yaml file to manually deploy DaemonSet.

        Run `kubectl get po -n kube-system -l k8s-app=logtail`. If any result is returned, you have manually deployed DaemonSet by using the yaml file.

     b. Download the latest DaemonSet template.

     c. Set *${your_region_name}*, *${your_aliyun_user_id}*, and *${your_machine_group_name}* as needed.

     d. Run `kubectl apply -f ./logtail-daemonset.yaml` to update the DaemonSet yaml file.

     For other comparison results, open a ticket.

Troubleshoot log collection exceptions in a container

If you cannot find any log on the preview or query page in the console, Log Service has not collected any log from your container. In this case, check the container status and perform the following steps:

1. Check whether the machine group status is normal.
2. Check whether the Config identifier is correct.

   Check whether IncludeLabel, ExcludeLabel, IncludeEnv, and ExcludeEnv in the Config match the configurations of the target container.

   > **Note** `Label` indicates the container label (label information in docker inspect) instead of the one defined in Kubernetes. You can temporarily remove the parameters and check whether any log can be collected. If yes, the exception is caused by an incorrect Config identifier.

3. Check other items.

If you want to collect files from your container, note that:

- Logtail does not collect any file if there are no modified files in your container.
- Only the files that are stored by default or mounted to your local PC can be collected.

Log on to the Logtail container

- **Common Docker**
  i. On the host, run `docker ps | grep logtail` to search for the Logtail container.
  ii. Run `docker exec -it ****** bash` to log on to the Logtail container.

```
$docker ps | grep logtail
223fbd3ed2a6e        registry.cn-hangzhou.aliyuncs.com/log-service/logtail
"/usr/local/ilogta..."   8 days ago          Up 8 days                                  logt
ail-iba
$docker exec -it 223fbd3ed2a6e bash
```

- **Kubernetes**
  i. Run `kubectl get po -n kube-system | grep logtail` to search for the Logtail Pod.
  ii. Run `kubectl exec -it -n kube-system ****** bash` to log on to the Pod.

```
$kubectl get po -n kube-system | grep logtail
logtail-ds-g5wgd                                          1/1       Running    0
8d
logtail-ds-slpn8                                          1/1       Running    0
8d
$kubectl exec -it -n kube-system logtail-ds-g5wgd bash
```

View Logtail operational logs

Logtail logs named *ilogtail.LOG* and *logtail_plugin.LOG* are stored in the */usr/local/ilogtail/* directory.

1. Log on to the Logtail container.
2. Open the */usr/local/ilogtail/* directory.

```
cd /usr/local/ilogtail
```

3. View the *ilogtail.LOG* and *logtail_plugin.LOG* files.

```
cat ilogtail.LOG
cat logtail_plugin.LOG
```

View Logtail standard output (stdout)

You can ignore the following stdout because the container stdout has no reference for application.

```
start umount useless mount points, /shm$|/merged$|/mqueue$
umount: /logtail_host/var/lib/docker/overlay2/3fd0043af174cb0273c3c7869500fbe2bdb95d13b1e11
0172ef57fe840c82155/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/d5b10aa19399992755de1f85d25009528daa749c1bf8c
16edff44beab6e69718/merged: must be superuser to unmount
umount: /logtail_host/var/lib/docker/overlay2/5c3125daddacedec29df72ad0c52fac800cd56c6e880d
c4e8a640b1e16c22dbe/merged: must be superuser to unmount
......
xargs: umount: exited with status 255; aborting
umount done
start logtail
ilogtail is running
logtail status:
ilogtail is running
```

View the status of log-related components in a Kubernetes cluster

To view the status of log-related components in a Kubernetes cluster, you can run `helm status alibaba-log-controller`.

View the version information, IP address, and time of Logtail

The related information is stored in the *app_info.json* file under the */usr/local/ilogtail/* directory in the Logtail container. The following is an example:

```
kubectl exec logtail-ds-gb92k -n kube-system cat /usr/local/ilogtail/app_info.json
{
   "UUID" : "",
   "hostname" : "logtail-gb92k",
   "instance_id" : "0EBB2B0E-0A3B-11E8-B0CE-0A58AC140402_172.20.4.2_1517810940",
   "ip" : "172.20.4.2",
   "logtail_version" : "0.16.2",
   "os" : "Linux; 3.10.0-693.2.2.el7.x86_64; #1 SMP Tue Sep 12 22:26:13 UTC 2017; x86_64",
   "update_time" : "2018-02-05 06:09:01"
}
```

What do I do if I mistakenly delete a Logstore that is created through CRD?

If you delete a Logstore that is automatically created through CRD, the collected data cannot be recovered, and the CRD configurations of the Logstore become invalid. In this case, you can use either of the following methods to prevent possible log collection exceptions:

- Use another CRD-created Logstore and take care to name the Logstore with a different name to the Logstore that was mistakenly deleted.
- Restart the *alibaba-log-controller* Pod. You can run `kubectl get po -n kube-system | grep alibaba-log-controller` to search for the Pod.

9.6. How do I obtain the labels and environment variables of a container?

Log Service allows you to collect logs from containers. You can specify the containers by label or environment variable. Labels are retrieved by running the docker inspect command and environment variables are specified in the startup configuration of each container.

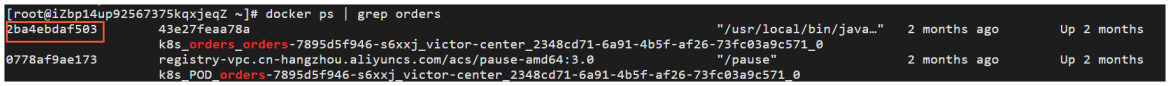
Obtain container labels

1. Log on to the host where the container whose labels you want to obtain resides, for example, an Elastic Compute Service (ECS) instance.
2. Run the following command to obtain the ID of the container.

   The *orders* variable in the command is the name of a container group. Replace the value of the variable with an actual name.

   ```
   docker ps | grep orders
   ```

   2ba4ebdaf503 in the response indicates the ID of the container.

3. Run the following command to obtain the labels of the container.

   The *2ba4ebdaf503* variable in the command is the ID of a container. Replace the value of the variable with an actual ID.

   ```
   docker inspect 2ba4ebdaf503
   ```

   The Labels field in the response indicates the container labels.

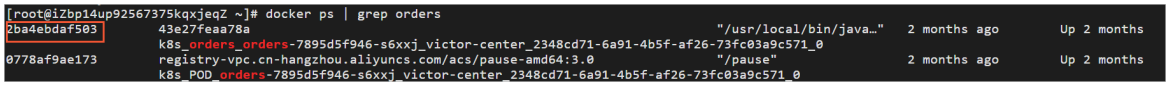
Obtain environment variables

1. Log on to the host where the container whose labels you want to obtain resides, for example, an Elastic Compute Service (ECS) instance.
2. Run the following command to obtain the ID of the container.

   The *orders* variable in the command is the name of a container group. Replace the value of the variable with an actual name.

   ```
   docker ps | grep orders
   ```

   2ba4ebdaf503 in the response indicates the ID of the container.

3. Run the following command to obtain the environment variables of the container.

The *2ba4ebdaf503* variable in the command is the ID of a container. Replace the value of the variable with an actual ID.

```
docker exec 2ba4ebdaf503 env
```

```
JAVA_OPTS=-Xms64m -Xmx128m -XX:PermSize=32m -XX:MaxPermSize=64m -XX:+UseG1GC -Djava.security.egd=file:/dev/urandom
FRONT_END_SERVICE_HOST=172.[redacted]225
PAYMENT_PORT_80_TCP=tcp://172.[redacted].5:80
CATALOGUE_DB_SERVICE_HOST=172.[redacted].96
MONGO_SERVICE_PORT_MONGO=27017
ANTICHEATING_PORT_80_TCP_PROTO=tcp
CATALOGUE_SERVICE_PORT_CATALOGUE=80
FRONT_END_PORT_8079_TCP_PROTO=tcp
FRONT_END_PORT_8079_TCP_ADDR=172.[redacted]25
USER_PORT_80_TCP_PROTO=tcp
MONGO_PORT_27017_TCP_ADDR=172.[redacted].199
PAYMENT_PORT=tcp://172.[redacted].5:80
CARTS_PORT_80_TCP=tcp://17[redacted]5.30:80
INTEGRAL_PORT=tcp://172.21.[redacted]3:80
CATALOGUE_PORT_80_TCP=tcp://172.[redacted]3:80
USER_PORT_80_TCP=tcp://172.[redacted].101:80
KUBERNETES_PORT=tcp://172.[redacted]443
CARTS_PORT=tcp://172.[redacted]5.30:80
TEST_PORT_27017_TCP_ADDR=172.[redacted].60
INTEGRAL_PORT_80_TCP_ADDR=17[redacted].12.53
CATALOGUE_SERVICE_HOST=17[redacted].133
SESSION_DB_PORT=tcp://172.[redacted]49:6379
INTEGRAL_SERVICE_PORT_INTEGRAL=80
RABBITMQ_PORT_5672_TCP_PORT=5672
RABBITMQ_PORT_5672_TCP_ADDR=172.[redacted].52
ORDERS_SERVICE_PORT=80
TEST_PORT_27017_TCP_PORT=27017
CATALOGUE_DB_PORT_3306_TCP=tcp://172.[redacted]6:3306
INTEGRAL_DB_SERVICE_PORT=3306
INTEGRAL_SERVICE_PORT=80
CARTS_SERVICE_HOST=172.[redacted].30
CARTS_SERVICE_PORT=80
CATALOGUE_PORT_80_TCP_PROTO=tcp
SESSION_DB_PORT_6379_TCP_PORT=6379
```

9.7. Query local collection status

Logtail is used to query its own health status and log collection progress, helping you troubleshoot log collection issues and customize status monitoring for log collection.

1. User guide
    i. all command
    ii. active command
    iii. logstore command
    iv. logfile command
    v. history command
2. Return values
3. Use cases
    i. Monitor the running status of Logtail
    ii. Monitor log collection progress
    iii. Determine whether or not Logtail has finished collecting log files
    iv. Troubleshoot log collection issues

User guide

If a Logtail client supporting status query function is installed, you can query local log collection status by entering commands on the client. To install Logtail, see Install Logtail on a Linux server.

Enter the `/etc/init.d/ilogtaild -h` command on the client to check if the client supports querying local log collection status. If the `logtail insight, version` keyword is returned, it indicates that this function is supported on the Logtail client.

```
/etc/init.d/ilogtaild -h
Usage: ./ilogtaild { start | stop (graceful, flush data and save checkpoints) | force-stop
| status | -h for help}$
logtail insight, version : 0.1.0
commond list :
       status all [index]
             get logtail running status
       status active [--logstore | --logfile] index [project] [logstore]
             list all active logstore | logfile. if use --logfile, please add project and l
ogstore. default --logstore
       status logstore [--format=line | json] index project logstore
             get logstore status with line or json style. default --format=line
       status logfile [--format=line | json] index project logstore fileFullPath
             get log file status with line or json style. default --format=line
       status history beginIndex endIndex project logstore [fileFullPath]
             query logstore | logfile history status.
index : from 1 to 60. in all, it means last $(index) minutes; in active/logstore/logfile/hi
story, it means last $(index)*10 minutes
```

Currently, Logtail supports the following query commands, command functions, time intervals to query and time windows for result statistics:

Command	Functions	Time interval to query	Time window for statistics
all	Query the running status of Logtail.	Last 60 min	1 min
active	Query Logstores or log files that are currently active (that is, with data collected).	Last 600 min	10 minutes.
logstore	Query the collection status of a Logstore.	Last 600 min	10 minutes.
logfile	Query the collection status of a log file.	Last 600 min	10 minutes.
history	Query the collection status of a Logstore or log file over a period of time.	Last 600 min	10 minutes.

> **Note**
>
> - The `index` parameter in the command represents the index value of the time window, which is counted from the current time. Its valid range is 1-60. If the time window for statistics is one minute, windows in the last `(index, index-1]` minutes are queried. If the time window for statistics is 10 minutes, windows in the last `(10*index, 10*(index-1)]` minutes are queried.
> - All query commands belong to status subcommands, so the main command is status.

all command

Command format

```
/etc/init.d/ilogtaild status all [ index ]
```

> **Note** The all command is used to view the running status of Logtail. The index parameter is optional. If left blank, 1 is taken by default.

Example

```
/etc/init.d/ilogtaild status all 1
ok
/etc/init.d/ilogtaild status all 10
busy
```

Output description

Item	Description	Priority	Resolution:
ok	The current status is normal.	None.	No action is needed.
busy	The current collection speed is high and the Logtail status is normal.	None.	No action is needed.
many_log_files	The number of logs being collected is large.	Low	Check if the configuration contains files that do not need to be collected.

Item	Description	Priority	Resolution:
process_block	Current log parsing is blocked.	Low	Check if logs are generated too quickly. If you still get this output, Configure the startup parameters of Logtail as per your needs to modify the upper limit of CPU usage or the limit on concurrent sending by using network.
send_block	Current sending is blocked.	Relatively high	blocked. Check if logs are generated too quickly and if the network status is normal. If you still get this output, Configure the startup parameters of Logtail as per your needs to modify the upper limit of CPU usage or the limit on concurrent sending by using network.
send_error	Failed to upload log data.	High	To troubleshoot the issue, see How do I view Logtail collection errors?.

active command

Command format

```
/etc/init.d/ilogtaild status active [--logstore] index
 /etc/init.d/ilogtaild status active --logfile index project-name logstore-name
```

Note

- The `active [--logstore] index` command is used to query Logstores that are currently active. The `--logstore` parameter can be omitted without changing the meaning of the command.
- The `active --logfile index project-name logstore-name` command is used to query all active log files in a Logstore for a project.
- The active command is used to query active log files level by level. We recommend that you first locate the currently active Logstore and then query active log files in this Logstore.

Example

```
/etc/init.d/ilogtaild status active 1
sls-zc-test : release-test
sls-zc-test : release-test-ant-rpc-3
sls-zc-test : release-test-same-regex-3
/etc/init.d/ilogtaild status active --logfile 1 sls-zc-test release-test
/disk2/test/normal/access.log
```

Output description

- To run the `active --logstore index` command, all currently active Logstores are output in the format of `project-name : logstore-name`. To run the `active --logfile index project-name logstore-name` command, the complete paths of active log files are output.
- A Logstore or log file with no log collection activity in the current query window does not appear in the output.

logstore command

Command format

```
/etc/init.d/ilogtaild status logstore [--format={line|json}] index project-name logstore-name
```

> **Note**
>
> - The logstore command is used to output the collection statuses of the specified project and Logstore in LINE or JSON format.
> - If the `--format=` parameter is not configured, `--format=line` is selected by default. The echo information is output in LINE format. Note that `--format` parameter must be placed behind `logstore`.
> - If this Logstore does not exist or has no log collection activity in the current query window, you get an empty output in LINE format or a `null` value in JSON format.

Example

```
/etc/init.d/ilogtaild status logstore 1 sls-zc-test release-test-same
time_begin_readable : 17-08-29 10:56:11
time_end_readable : 17-08-29 11:06:11
time_begin : 1503975371
time_end : 1503975971
project : sls-zc-test
logstore : release-test-same
status : ok
config : ##1.0##sls-zc-test$same
read_bytes : 65033430
parse_success_lines : 230615
parse_fail_lines : 0
last_read_time : 1503975970
read_count : 687
avg_delay_bytes : 0
max_unsend_time : 0
min_unsend_time : 0
max_send_success_time : 1503975968
send_queue_size : 0
send_network_error_count : 0
send_network_quota_count : 0
send_network_discard_count : 0
send_success_count : 302
send_block_flag : false
sender_valid_flag : true
/etc/init.d/ilogtaild status logstore --format=json 1 sls-zc-test release-test-same
{
   "avg_delay_bytes" : 0,
   "config" : "##1.0##sls-zc-test$same",
   "last_read_time" : 1503975970,
   "logstore" : "release-test-same",
   "max_send_success_time" : 1503975968,
   "max_unsend_time" : 0,
   "min_unsend_time" : 0,
   "parse_fail_lines" : 0,
   "parse_success_lines" : 230615,
   "project" : "sls-zc-test",
   "read_bytes" : 65033430,
   "read_count" : 687,
   "send_block_flag" : false,
   "send_network_discard_count" : 0,
   "send_network_error_count" : 0,
   "send_network_quota_count" : 0,
   "send_queue_size" : 0,
   "send_success_count" : 302,
   "sender_valid_flag" : true,
   "status" : "ok",
   "time_begin" : 1503975371,
   "time_begin_readable" : "17-08-29 10:56:11",
   "time_end" : 1503975971,
   "Maid": "17-08-29 11:06:11"
}
```

Output description

Reserved Word	Meaning	Unit
Status	The overall status of this Logstore. For specific statuses, descriptions, and change methods, see the following table.	None.
time_begin_readable	The start time that can be read.	None.
time_end_readable	The end time that can be read.	None.
time_begin	The start time of statistics.	UNIX timestamp, measured in seconds.
time_end	The end time of statistics.	UNIX timestamp, measured in seconds.
project	The project name.	None.
logstore	The Logstore name.	None.
config	The collection configuration name, which is globally unique and consisted of `##1.0##`, project, `$`, and config.	None.
read_bytes	The number of logs read in the window.	Byte
parse_success_lines	The number of successfully parsed log lines in the window.	Line
parse_fail_lines	The number of log lines that failed to be parsed in the window.	Line
last_read_time	The last read time in the window.	UNIX timestamp, measured in seconds.
Read_count	The number of times that logs are read in the window.	Number
avg_delay_bytes	The average of the differences between the current offset and the file size each time logs are read in the window.	Byte
max_unsend_time	The maximum time that unsent data packets are in the send queue when the window ends. The value is 0 when the queue is empty.	UNIX timestamp, measured in seconds.

Reserved Word	Meaning	Unit
min_unsend_time	The minimum time that unsent data packets are in the send queue when the window ends. The value is 0 when the queue is empty.	UNIX timestamp, measured in seconds.
max_send_success_time	The maximum time that data is successfully sent in the window.	UNIX timestamp, measured in seconds.
send_queue_size	The number of unsent data packets in the current send queue when the window ends.	Packet
send_network_error_count	The number of data packets that failed to be sent in the window because of network errors.	Packet
send_network_quota_count	The number of data packets that failed to be sent in the window because the quota is exceeded.	Packet
send_network_discard_count	The number of discarded data packets in the window because of data exceptions or insufficient permissions.	Packet
send_success_count	The number of successfully sent data packets in the window.	Packet
send_block_flag	Whether or not the send queue is blocked when the window ends.	None.
sender_valid_flag	Whether or not the send flag of this Logstore is valid when the window ends. true means the flag is valid, and false means the flag is disabled because of network errors or quota errors.	None.

Logstore status

Status	Meaning	Handling method
ok	The status is normal.	No action is needed.

Status	Meaning	Handling method
process_block	Log parsing is blocked.	Check if logs are generated too quickly. If you still get this output, Configure Configure the startup parameters of Logtail as per your needs to modify the upper limit of CPU usage or the limit on concurrent sending by using network.
parse_fail	Log parsing failed.	Check whether or not the log format is consistent with the log collection configuration.
send_block	Current sending is blocked.	blocked. Check if logs are generated too quickly and if the network status is normal. If you still get this output, Configure the startup parameters of Logtail as per your needs to modify the upper limit of CPU usage or the limit on concurrent sending by using network.
sender_invalid	An exception occurred when sending log data.	Check the network status. If the network is normal, see How do I view Logtail collection errors? in Query diagnosis errors to troubleshoot the issue.

logfile command

Command format

```
/etc/init.d/ilogtaild status logfile [--format={line|json}] index project-name logstore-nam
e fileFullPath
```

Note

- The logfile command is used to output the collection status of a specific log file in LINE or JSON format.
- If the `--format=` parameter is not configured, `--format=line` is selected by default. The echo information is output in LINE format.
- If this log file does not exist or has no log collection activity in the current query window, you get an empty output in LINE format or a `null` value in JSON format.
- The `--format` parameter must be placed behind `logfile` .
- The `filefullpath` must be a full path name.

Example

```
/etc/init.d/ilogtaild status logfile 1 sls-zc-test release-test-same /disk2/test/normal/acc
ess.log
time_begin_readable : 17-08-29 11:16:11
time_end_readable : 17-08-29 11:26:11
time_begin : 1503976571
time_end : 1503977171
project : sls-zc-test
logstore : release-test-same
status : ok
config : ##1.0##sls-zc-test$same
file_path : /disk2/test/normal/access.log
file_dev : 64800
file_inode : 22544456
file_size_bytes : 17154060
read_offset_bytes : 17154060
read_bytes : 65033430
parse_success_lines : 230615
parse_fail_lines : 0
last_read_time : 1503977170
read_count : 667
avg_delay_bytes : 0
/etc/init.d/ilogtaild status logfile --format=json 1 sls-zc-test release-test-same /disk2/t
est/normal/access.log
{
   "avg_delay_bytes" : 0,
   "config" : "##1.0##sls-zc-test$same",
   "file_dev" : 64800,
   "file_inode" : 22544456,
   "file_path" : "/disk2/test/normal/access.log",
   "file_size_bytes" : 17154060,
   "last_read_time" : 1503977170,
   "logstore" : "release-test-same",
   "parse_fail_lines" : 0,
   "parse_success_lines" : 230615,
   "project" : "sls-zc-test",
   "read_bytes" : 65033430,
   "read_count" : 667,
   "read_offset_bytes" : 17154060,
   "status" : "ok",
   "time_begin" : 1503976571,
   "time_begin_readable" : "17-08-29 11:16:11",
   "time_end" : 1503977171,
   "time_end_readable" : "17-08-29 11:26:11"
}
```

Output description

Reserved Word	Meaning	Unit

Reserved Word	Meaning	Unit
Status	The collection status of this log file in the current query window. See the status of logstore command.	None.
time_begin_readable	The start time that can be read.	None.
time_end_readable	The end time that can be read.	None.
time_begin	The start time of statistics.	UNIX timestamp, measured in seconds.
time_end	The end time of statistics.	UNIX timestamp, measured in seconds.
project	The project name.	None.
logstore	The Logstore name.	None.
file_path	The path of the log file.	None.
file_dev	The device ID of the log file.	None.
file_inode	The inode of the log file.	None.
file_size_bytes	The size of the last scanned file in the window.	Byte
read_offset_bytes	The parsing offset of this file.	Byte
config	The collection configuration name, which is globally unique and consisted of `##1.0##` , project, `$` and config.	None.
read_bytes	The number of logs read in the window.	Byte
parse_success_lines	The number of successfully parsed log lines in the window.	Line
parse_fail_lines	The number of log lines that failed to be parsed in the window.	Line
last_read_time	The last read time in the window.	UNIX timestamp, measured in seconds.
read_count	The number of times that logs are read in the window.	Number of times

Reserved Word	Meaning	Unit
avg_delay_bytes	The average of the differences between the current offset and the file size each time logs are read in the window.	Byte

history command

Command format

```
/etc/init.d/ilogtaild status history beginIndex endIndex project-name logstore-name [fileFullPath]
```

> **Note**
>
> - The history command is used to query the collection status of a Logstore or log file over a period of time.
> - `beginIndex` and `endIndex` represent the start and end values for the code query window index respectively. `beginIndex <= endIndex` .
> - If the `fileFullPath` is not entered in the parameter, the code queries the collection information of the Logstore. Otherwise, the collection information of the log file is queried.

Example

```
/etc/init.d/ilogtaild status history 1 3 sls-zc-test release-test-same /disk2/test/normal/access.log
       begin_time status read parse_success parse_fail      last_read_time read_count avg_delay device inode file_size read_offset
 17-08-29 11:26:11 ok 62.12MB 231000 0 17-08-29 11:36:11 671 0B 64800 22544459 18.22MB 18.22MB
 17-08-29 11:16:11 ok 62.02MB 230615 0 17-08-29 11:26:10 667 0B 64800 22544456 16.36MB 16.36MB
 17-08-29 11:06:11 ok 62.12MB 231000 0 17-08-29 11:16:11 687 0B 64800 22544452 14.46MB 14.46MB
$/etc/init.d/ilogtaild status history 2 5 sls-zc-test release-test-same
       begin_time status read parse_success parse_fail      last_read_time read_count avg_delay send_queue network_error quota_error discard_error send_success send_block send_valid max_unsend min_unsend    max_send_success
 17-08-29 11:16:11 ok 62.02MB 230615 0 17-08-29 11:26:10 667 0B 0 0 0 0 300 false true 70-01-01 08:00:00 70-01-01 08:00:00 17-08-29 11:26:08
 17-08-29 11:06:11 ok 62.12MB 231000 0 17-08-29 11:16:11 687 0B 0 0 0 0 303 false true 70-01-01 08:00:00 70-01-01 08:00:00 17-08-29 11:16:10
 17-08-29 10:56:11 ok 62.02MB 230615 0 17-08-29 11:06:10 687 0B 0 0 0 0 302 false true 70-01-01 08:00:00 70-01-01 08:00:00 17-08-29 11:06:08
 17-08-29 10:46:11 ok 62.12MB 231000 0 17-08-29 10:56:11 692 0B 0 0 0 0 302 false true 70-01-01 08:00:00 70-01-01 08:00:00 17-08-29 10:56:10
```

Output description

- This command outputs historical collection information of a Logstore or log file in the form of list, one line for each window.
- For the description of each output field, see the `logstore` and `logfile` commands.

Return values

Normal return value

0 is returned if a command input is valid (including failure to query a Logstore or log file), for example:

```
/etc/init.d/ilogtaild status logfile --format=json 1 error-project error-logstore /no/this/
file
null
echo $?
0
/etc/init.d/ilogtaild status all
ok
echo $?
0
```

Exceptional return values

A non-zero return value indicates an exception. See the following table.

Return value	Type	output	Troubleshooting
10	Invalid command or missing parameters	`invalid param, use -h for help.`	Enter `-h` to view help.
1	The query goes beyond the 1-60 time window	`invalid query interval`	Enter `-h` to view help.
1	Cannot query the specified time window	`query fail, error: $(error)` . For more information, see errno interpretation.	This issue might occur when the startup time of Logtail is less than the query time span. For other cases, open a ticket.
1	No matching query window time	`no match time interval, please check logtail Status`	Check if Logtail is running. For other cases, open a ticket.
1	No data in the query window	`invalid profile, maybe logtail Restart`	Check if Logtail is running. For other cases, open a ticket.

Example

```
/etc/init.d/ilogtaild status nothiscmd
invalid param, use -h for help.
echo $?
10
/etc/init.d/ilogtaild status/all 99
invalid query interval
echo $?
1
```

Use cases

You can obtain the overall status of Logtail by querying its health status, and obtain the related metrics during collection by querying the collection progress. With the obtained information, you can monitor log collection in a customized manner.

Monitor the running status of Logtail

Monitor the running status of Logtail by using the `all` command.

How it works: The current status of Logtail is queried every minute. If Logtail is under `process_block`, `send_block`, or `send_error` status for five successive minutes, an alarm is triggered.

The alarm duration and the status range being monitored can be adjusted according to the importance of log collection in specific scenarios.

Monitor log collection progress

Monitor the collection progress of a Logstore by using the `logstore` command.

How it works: The `logstore` command is called every ten minutes to obtain the status information of this Logstore. If the `avg_delay_bytes` is over 1 MB (1024*1024) or `status` is not `ok`, an alarm is triggered.

The `avg_delay_bytes` alarm threshold can be adjusted according to the log collection traffic.

Determine whether or not Logtail has finished collecting log files

Determine whether or not Logtail has finished collecting log files by using the `logfile` command.

How it works: After writing to the log file stops, the `logfile` command is called every ten minutes to obtain the status information of this file. If this file shows the same value for `read_offset_bytes` and `file_size_bytes`, it means that Logtail has finished collecting this log file.

Troubleshoot log collection issues

If the log collection is delayed on a server, use the `history` command to query related collection information on this server.

1. If the `send_block_flag` is true, it indicates that the log collection delays because of the network.
   - If the `send_network_quota_count` is greater than 0, you must split the Shard of the Logstore.
   - If the `send_network_error_count` is greater than 0, you must check the network connectivity.
   - ◦If no related network error occurs, you must adjust the limit on concurrent sending and traffic limit of Logtail.

2. Sending-related parameters are normal, but the `avg_delay_bytes` is relatively high.
   - The average log parsing speed can be calculated by using `read_bytes` to determine if traffic generated by logs is normal.
   - Resource usage limits of Logtail can be adjusted as appropriate.
3. The `parse_fail_lines` is greater than 0.

   Check if the parsing configurations for log collection match with all the logs.

9.8. What do I do if errors occur when I use Logtail to collect logs?

If the preview page is blank or the query page displays no data when you use Logtail to collect logs, perform the following steps to troubleshoot the errors:

Procedure

1. Check whether the heartbeat status of your machine group is normal.

   View the heartbeat status of your machine group in the Log Service console. For more information, see View the status of a machine group.
   - If a value in the Heartbeat column is **FAIL**, troubleshoot the error by following the instructions that are provided in What do I do if a Logtail machine group has no heartbeats?.
   - If all values in the Heartbeat column are **OK**, proceed to the next step.
2. Check whether a Logtail configuration is created.
   - If no Logtail configuration is created, create a Logtail configuration by following the instructions that are provided in Create Logtail configurations.
   - If a Logtail configuration is created, proceed to the next step.

   > **Notice** Make sure that the log path specified in the Logtail configuration matches the log files on the servers from which you want to collect logs.
3. Check whether the Logtail configuration is applied to your machine group.

   On the **Machine Group Settings** page, check whether the Logtail configuration is applied to your machine group. For more information, see Manage machine groups.
   - If the Logtail configuration is not applied to your machine group, apply the Logtail configuration by following the instructions that are provided in Manage Logtail configurations.
   - If the Logtail configuration is applied to your machine group, proceed to the next step.
4. View collection errors.

   Check whether new logs are generated in the log files in real time. Logtail collects only incremental logs. If the log files are not updated, Logtail does not read the log files. If the log files are updated but no results are returned when you query the updated data in Log Service, perform the following operations:
   - View collection errors.

     For more information, see How do I view Logtail collection errors?.
   - View the logs of Logtail.

Logtail records important information and all WARNING and ERROR logs. If you want to view the details of the errors, you can access the logs of Logtail in the following paths:

- Linux: */usr/local/ilogtail/ilogtail.LOG* and */usr/local/ilogtail/logtail_plugin.LOG*. The files contain logs that are collected when the Logtail configuration uses a data source such as HTTP, MySQL Binlog, or MySQL query results.
- Windows x64: *C:\Program Files (x86)\Alibaba\Logtail\logtail_*.log*.
- Windows x32: *C:\Program Files\Alibaba\Logtail\logtail_*.log*.

- Check whether limits are exceeded.

  If you want to collect a large amount of log data or collect logs from a large number of files, you can modify the startup parameters of Logtail to increase the throughput for log collection. For more information, see Configure the startup parameters of Logtail.

If the errors persist after you perform the preceding operations, submit a ticket and provide important information that you found in troubleshooting.

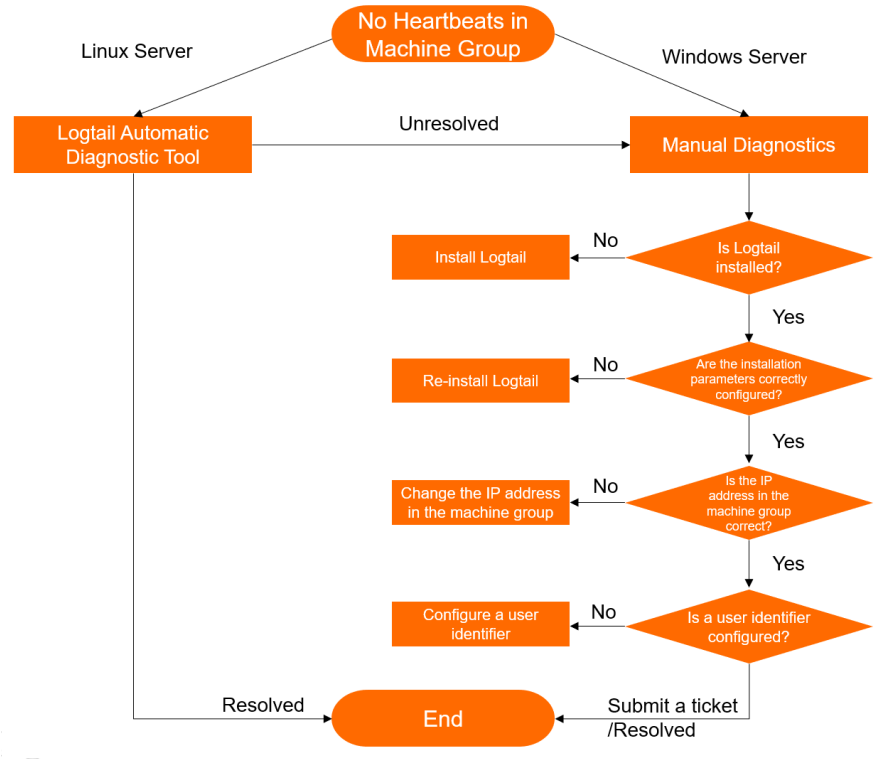
9.9. What do I do if a Logtail machine group has no heartbeats?

If no heartbeats are detected in a Logtail machine group when you use Logtail to collect logs, you can troubleshoot the error manually or by using the Logtail automatic diagnostic tool. A Logtail machine group contains servers on which Logtail is installed. This topic describes how to troubleshoot the error if a Logtail machine group has no heartbeats.

Troubleshooting process

If you use Logtail to collect logs, Logtail sends heartbeat packets to Log Service at a scheduled time after Logtail is installed on a server. If no heartbeats are detected in the machine group to which the server belongs, the connection between Logtail and Log Service fails. Log Service provides automatic and manual diagnostic methods. You can select a method based on your business requirements.

- Automatic diagnostics: Log Service provides the Logtail automatic diagnostic tool. You can select this method only if Logtail is installed on Linux servers. For more information, see How do I use the Logtail automatic diagnostic tool?
- Manual diagnostics: If the Logtail automatic diagnostic tool fails to identify the root cause of the error or Logtail is installed on a Windows server, perform the following steps.



Step 1: Check whether Logtail is installed

View the status of Logtail to check whether Logtail is installed on a server.

- Linux:

  Run the following command to view the status of Logtail:

  ```
  sudo /etc/init.d/ilogtaild status
  ```

  If the following information is returned, Logtail is installed:

  ```
  ilogtail is running
  ```

- Windows:

  i. Open the Run window and enter `services.msc` to open the Services window.

  ii. View the status of the LogtailDaemon and LogtailWorker services.

  If the services are in the Running state, Logtail is installed.

Proceed based on the check results.

- If Logtail is not installed, install Logtail by following the instructions that are provided in Install Logtail on a Linux server or Install Logtail on a Windows server.

  Make sure that you install Logtail based on the region where your Log Service project resides and the network type that is used for log collection. For more information about network types, see Select a network type.

- If Logtail is installed, proceed to the next step.

Step 2: Check whether the Logtail installation parameters are correctly configured

When you install Logtail, you must specify a correct Log Service endpoint for Logtail to connect to Log Service. You must enter the name of the region where your project resides and select the installation method based on the network type. For more information about region names, see Region names for Logtail installation. For more information about network types, see Select a network type. If the installation parameters are incorrectly configured or the Logtail installation script is invalid, heartbeats may not be detected on the server on which Logtail is installed. For more information about Log Service endpoints for different regions, see Endpoints.

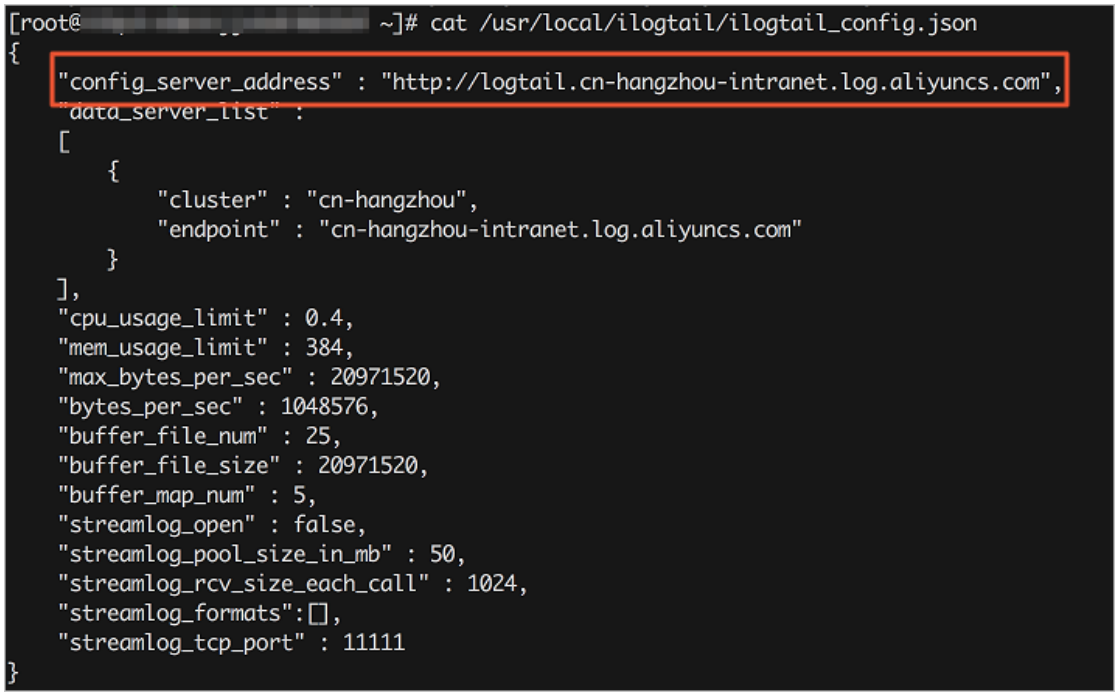
The Logtail configuration file *ilogtail_config.json* contains the Logtail installation parameters and the installation method. The following list describes the paths to the file:

- Linux: */usr/local/ilogtail/ilogtail_config.json*
- 64-bit Windows: *C:\Program Files (x86)\Alibaba\Logtail\ilogtail_config.json*
- 32-bit Windows: *C:\Program Files\Alibaba\Logtail\ilogtail_config.json*

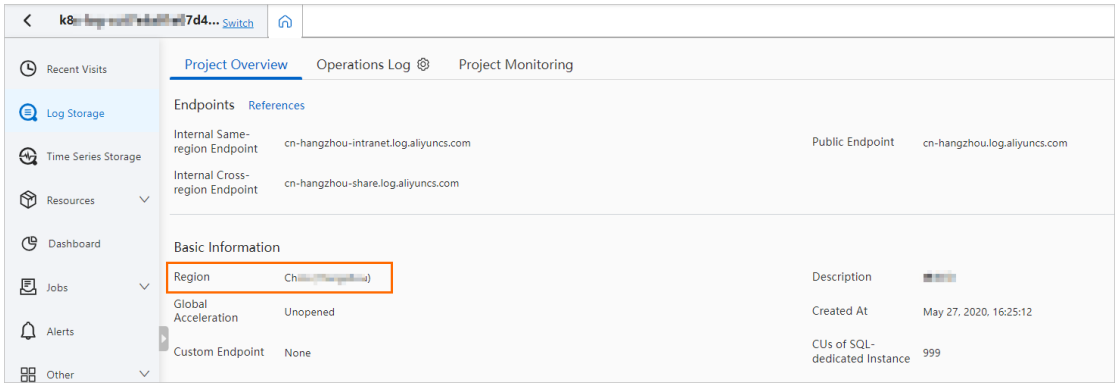
1. Check whether the region of the Log Service endpoint that is used by Logtail is the same as the region of your project in the *ilogtail_config.json* file.
    i. Run the following command on your server to view the region of the Log Service endpoint that is used by Logtail:

    ```
    cat /usr/local/ilogtail/ilogtail_config.json
    ```

    The following information is returned, which indicates that Logtail is installed on an Elastic Compute Service (ECS) instance in the China (Hangzhou) region.

ii. View the region of your project in the Log Service console.



2. View the endpoint that is specified in the *ilogtail_config.json* file and check whether the selected installation method is correct based on the network type of your server.

   For example, the endpoint `cn-hangzhou-intranet.log.aliyuncs.com` is specified in the *ilogtail_config.json* file.

   - Linux:

     Run the following command to test the network connectivity:

     ```
     curl logtail.cn-hangzhou-intranet.log.aliyuncs.com
     ```

     If information similar to the following example is returned, the network is connected:

     ```
     null
     ```

   - Windows:

     Run the following command to test the network connectivity:

     ```
     telnet logtail.cn-hangzhou-intranet.log.aliyuncs.com 80
     ```

     If information similar to the following example is returned, the network is connected:

     ```
     Trying 100*0*7*5...
     Connected to logtail.cn-hangzhou-intranet.log.aliyuncs.com.
     Escape character is '^]'.
     ```

- If the check fails, the installation parameters are incorrectly configured. The system displays a message indicating that an incorrect installation command is used. In this case, you must reconfigure the installation parameters. For more information, see Install Logtail on a Linux server or Install Logtail on a Windows server.
- If the installation parameters are correctly configured, proceed to the next step.

Step 3: Check whether the IP address that is specified in the machine group is correct

The server IP address that is obtained by Logtail must be the same as the IP address that is specified in the machine group. Otherwise, the machine group has no heartbeats, or logs cannot be collected. Logtail obtains a server IP address by using the following methods:

- If the hostname of the server is not bound to an IP address, Logtail obtains the IP address of the first network interface controller (NIC) card of the server.

- If the hostname of the server is bound to an IP address, Logtail obtains the IP address. You can view the hostname and IP address in the */etc/hosts* file.

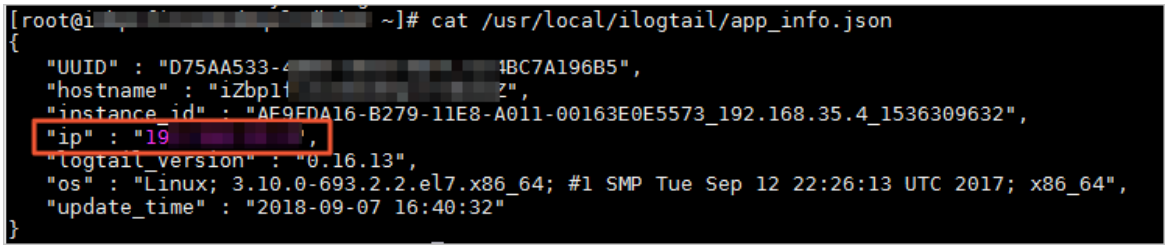
  > **Note** You can obtain a hostname from the **hostname** field.

  1. View the server IP address that is obtained by Logtail.

     The ip field in the *app_info.json* file records the server IP address that is obtained by Logtail. The following list describes the paths to the file:

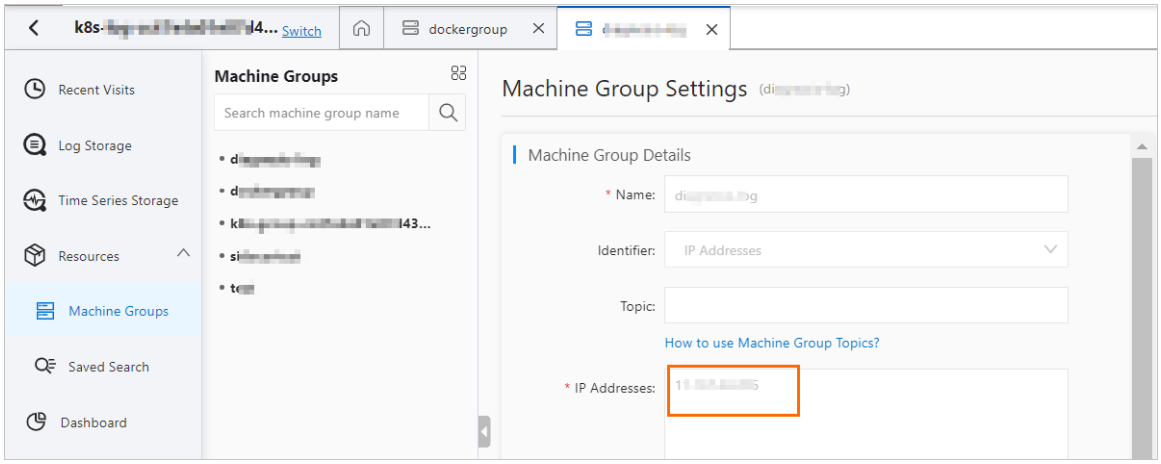
     - Linux: */usr/local/ilogtail/app_info.json*
     - 64-bit Windows: *C:\Program Files (x86)\Alibaba\Logtail\app_info.json*
     - 32-bit Windows: *C:\Program Files\Alibaba\Logtail\app_info.json*

     > **Notice**
     > - If the ip field in the `app_info.json` file is empty, Logtail cannot work. In this case, you must configure an IP address for the server and restart Logtail.
     > - The `app_info.json` file is used only to record information. If you specify the IP address in the file, the server IP address that is obtained by Logtail is not updated.

  2. View the IP address that is specified in the machine group.

     For more information, see Manage machine groups.

- If the IP address that is specified in the machine group is different from the IP address that is obtained by Logtail, change the IP address in the machine group.
  - If the IP address that is specified in the machine group is incorrect, change the IP address in the machine group. Wait 1 minute before you check the heartbeat status of the machine group.

- If you change the network configurations of the server on which Logtail is installed, restart Logtail to update the IP address that is obtained by Logtail. For example, you can modify the */etc/hosts* file to change the network configurations. Then, change the IP address in the machine groups to the value of the `ip` field in the `app_info.json` file. You can restart Logtail by using the following methods:
  - Linux:

    ```
    sudo /etc/init.d/ilogtaild stop
    sudo /etc/init.d/ilogtaild start
    ```

  - Windows:
    a. Open the Run window and enter `services.msc` to open the Services window.
    b. Restart the LogtailWorker service.

- If the IP address that is specified in the machine group is the same as the server IP address that is obtained by Logtail, proceed to the next step.

Step 4: Check whether a user identifier is configured

If your server is an ECS instance that belongs to a different Alibaba Cloud account than Log Service, a server from a third-party cloud service provider, or an on-premises server in a data center, you must specify the ID of the Alibaba Cloud account to which Log Service belongs as a user identifier for your server after you install Logtail on your server. This way, Logtail is authorized to collect logs from your server across different accounts. For more information, see Configure a user identifier.

Check whether a file named after the ID of the Alibaba Cloud account to which Log Service belongs exists in the */etc/ilogtail/users* directory.

- If yes, the user identifier is configured.
- If no, configure a user identifier. For more information, see Configure a user identifier.

> **Notice** The user identifier must be the ID of an Alibaba Cloud account. For more information, see Obtain the ID of the Alibaba Cloud account to which Log Service belongs.

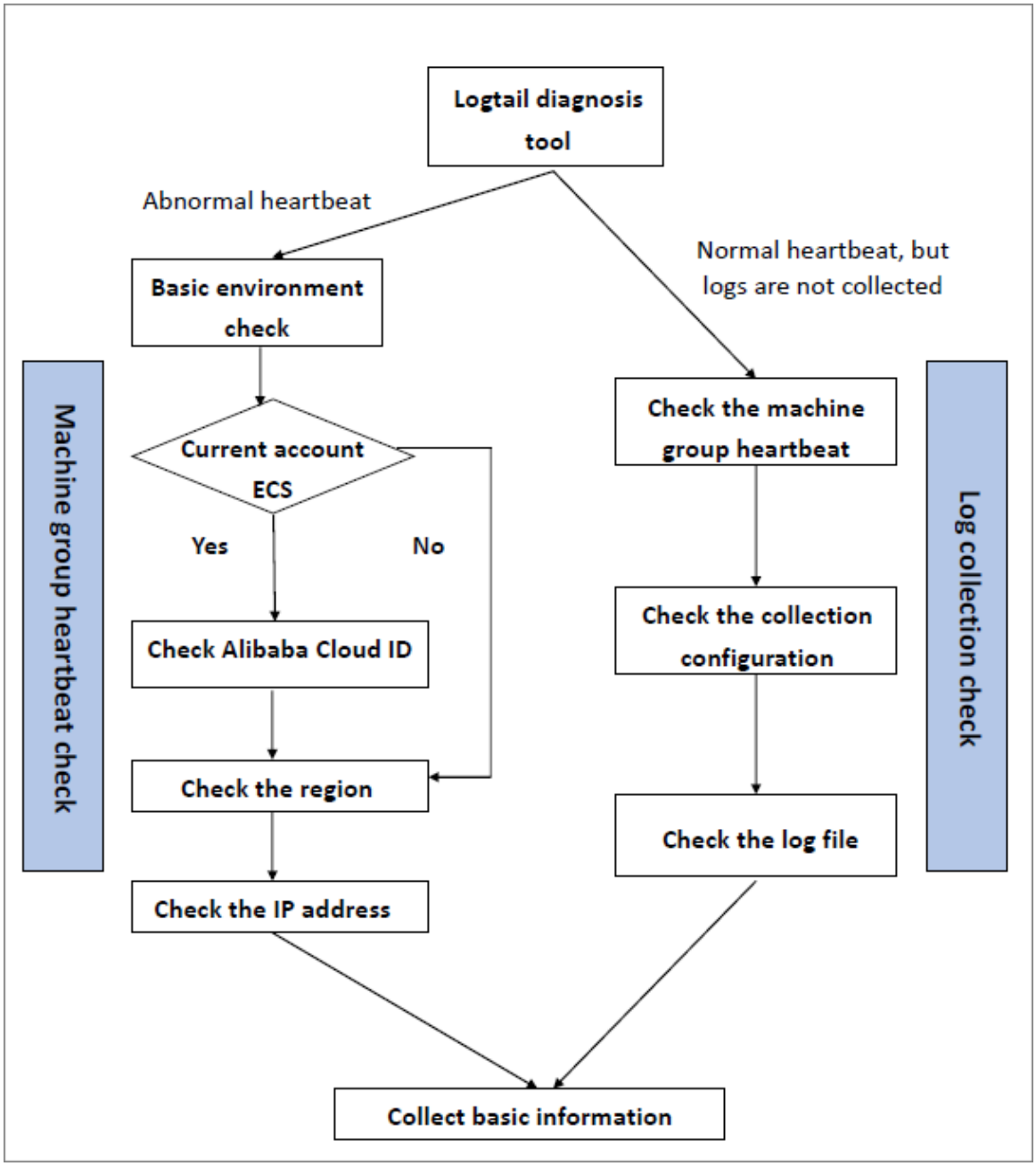
If the issue persists after you perform the preceding operations, submit a ticket and provide the information about your project, Logstore, and machine group. In addition, provide the app_info.json and ilogtail_config.json files as well as the output of the Logtail automatic diagnostic tool.

9.10. How do I use the automatic diagnostic tool of Logtail?

If an error occurs when you use Logtail to collect logs, you can use the automatic diagnostic tool of Logtail to check whether the error occurs on the Logtail client. This way, you can efficiently identify and fix the error.

> **Notice** The automatic diagnostic tool of Logtail is available only for Linux servers.

Diagnostic process



Download and run the diagnostic tool

1. Log on to a Linux server.
2. Download the diagnostic tool script.

   ```
   wget http://logtail-release.oss-cn-hangzhou.aliyuncs.com/linux64/checkingtool.sh -O checkingtool.sh
   ```

   If you cannot download the diagnostic tool script by running the preceding command, run the following command to download the script from the secondary address:

   ```
   wget http://logtail-corp.oss-cn-hangzhou-zmf.aliyuncs.com/linux64/checkingtool.sh -O checkingtool.sh
   ```

3. Install the curl tool.

The automatic diagnostic tool of Logtail uses the curl tool to check the network connectivity. Make sure that the curl tool is installed on the Linux server.

4. Run the diagnostic tool.

```
chmod 744 ./checkingtool.sh
./checkingtool.sh
sh checkingtool.sh
```

Information similar to the following example is returned:

```
[Info]:     Logtail checking tool version : 0.3.0
[Input]:  please choose which item you want to check :
            1. MachineGroup heartbeat fail.
            2. MachineGroup heartbeat is ok, but log files have not been collected.
Item :
```

5. Enter `1` or `2` as prompted. The script performs different checks based on the option that you select.
   - `1`: The script checks the heartbeat status of the machine group. If the heartbeat status of the machine group is FAIL, select this option.
   - `2`: The script checks whether errors occur when Logtail collects logs. If the heartbeat status of the machine group is OK but no log file is collected, select this option.

Check the heartbeat status of the machine group

If you enter 1 to check the issue that Log Service cannot receive heartbeats from the Logtail client, the automatic diagnostic tool of Logtail performs the following checks:

1. Check whether the basic environment is stable.
   - Check whether Logtail is installed.
   - Check whether Logtail is running.
   - Check whether the status of Secure Sockets Layer (SSL) encryption is normal.
   - Check whether the network connection between Logtail and Log Service is normal.

```
[Info]:     Logtail checking tool version : 0.3.0
[Input]:  please choose which item you want to check :
               1. MachineGroup heartbeat fail.
               2. MachineGroup heartbeat is ok, but log files have not been collected.
        Item : 1
[Info]:     Check logtail install files
[Info]:     Install file: ilogtail_config.json exists.                          [ OK ]
[Info]:     Install file: /etc/init.d/ilogtaild exists.                         [ OK ]
[Info]:     Install file: ilogtail exists.                                      [ OK ]
[Info]:     Bin file: /usr/local/ilogtail/ilogtail_0.14.2 exists.               [ OK ]
[Info]:     Logtail version :                                                   [ OK ]
[Info]:     Check logtail running status
[Info]:     Logtail is runnings.                                                [ OK ]
[Info]:     Check network status
[Info]:     Logtail is using ip: 11.XX.XX.187
[Info]:     Logtail is using UUID: 0DF18E97-0F2D-486F-B77F-XXXXXXXXXXXX
[Info]:     Check SSL status
[Info]:     SSL status OK.                                                      [ OK ]
[Info]:     Check logtail config server
[Info]:     config server address: http://config.sls.aliyun-inc.com
[Info]:     Logtail config server OK                                            [ OK ]
```

If an `Error` message appears, fix the error as prompted.

2. Check whether the server is an Elastic Compute Service (ECS) instance of the current Alibaba Cloud account.

   ```
   [Input]: Is your server non-Alibaba Cloud ECS or not belong to the same account with th
   e current Project of Log Service ? (y/N)
   ```

   - If the server is an ECS instance of the current Alibaba Cloud account, enter `N`.
   - If the server is an ECS instance that belongs to another Alibaba Cloud account, a server that is provided by a third-party cloud service provider, or a self-managed data center, enter `y`.

     If you enter `y`, the diagnostic tool returns the information about the locally configured user identifiers. Check whether the ID of your Alibaba Cloud account is included. If the ID of your Alibaba Cloud account is not included, configure a user identifier. For more information, see Configure a user identifier.

     ```
     [Input]:  Is your server non-Alibaba Cloud ECS or not belong to the same account with
     the current Project of Log Service ? (y/N)y
     [Info]:     Check aliyun user id(s)
     [Info]:     aliyun user id : 126XXXXXXXXX79 .                                [ OK ]
     [Info]:     aliyun user id : 165XXXXXXXXX50 .                                [ OK ]
     [Info]:     aliyun user id : 189XXXXXXXXX57 .                                [ OK ]
     [Input]:  Is your project owner account ID is the above IDs ? (y/N)
     ```

3. Check whether the region where your project resides is the same as the region that you selected when you install Logtail.

   ```
   [Input]: please make sure your project is in this region : { cn-hangzhou } (y/N) :
   ```

   If the regions are different, reinstall Logtail. For more information, see Install Logtail on a Linux server.

4. Check whether the IP address or custom ID in your machine group is the same as the IP address or

custom ID that is displayed in the returned message.

```
[Input]:  please make sure your machine group's ip is same with : { 11.XX.XX.187 } or y
our machine group's userdefined-id is in : { XX-XXXXX } (y/N) :
```

If the IP addresses or custom IDs are different, modify the IP address or custom ID in the machine group. For more information, see Modify a machine group.

Check whether errors occur when Logtail collects logs

If you enter 2 to check whether errors occur when Logtail collects logs, the automatic diagnostic tool of Logtail performs the following checks:

1. Check whether the IP address in your machine group is the same as the IP address that is displayed in the returned message.

   ```
   [Input]:  please make sure your machine group's ip is same with : { 11.XX.XX.187 } (y/N
   ) :
   ```

   If the IP addresses are different, modify the IP address in the machine group. For more information, see Modify a machine group.

2. Check whether your Logtail configuration is applied to the machine group.

   ```
   [Input]: please make sure you have applied collection config to the machine group (y/N)
   :Y
   ```

   If the Logtail configuration is not applied to the machine group, apply the Logtail configuration to the machine group. For more information, see Manage Logtail configurations.

3. Check whether the log file in the Logtail configuration is valid.

   You must enter the full path of the log file that you want to check. If no file is matched, check whether the specified path can match the file.

   If the log file path is invalid, modify the Logtail configuration, and then run the script to repeat the check after 1 minute. For more information about how to modify a Logtail configuration, see Modify Logtail configurations.

   ```
   [Input]:  please input your log file's full path (eg. /var/log/nginx/access.log) :/disk
   2/logs/access.log
   [Info]:     Check specific log file
   [Info]:     Check if specific log file [ /disk2/logs/access.log ] is included by user c
   onfig.
   [Warning]:  Specific log file doesnt exist.                                          [ Warni
   ng ]
   [Info]:     Matched config found:                                                    [ OK ]
   [Info]:     [Project] -> sls-zc-xxxxxx
   [Info]:     [Logstore] -> release-xxxxxxx
   [Info]:     [LogPath] -> /disk2/logs
   [Info]:     [FilePattern] -> *.log
   ```

Submit a ticket if the issue still persists when all checks are passed

If the Logtail client passes all checks but still cannot collect logs, enter `y` for the last option in the script and press Enter. Information similar to the following example is returned:

```
[Input]: please make sure all the check items above have passed. If the problem persists, p
lease copy all the outputs and submit a ticket in the ticket system. : (y/N)y
```

Press y to submit a ticket. You must include the output of the check script in the ticket.

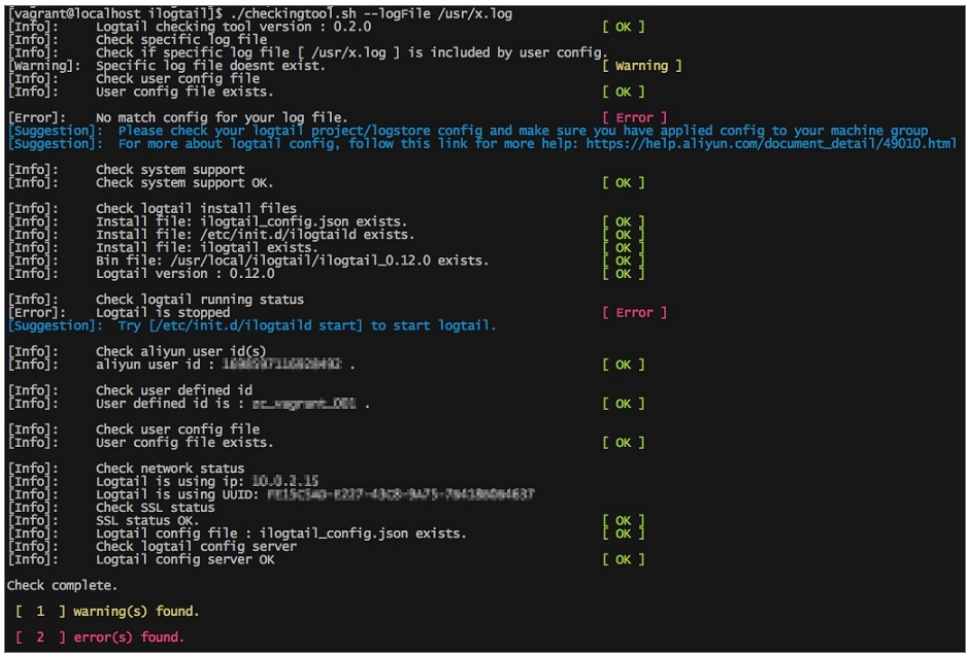
Run a quick check

You can run a quick check without confirmation. You can encapsulate and customize a quick check script.

> **Note** During a quick check, the diagnostic tool returns the user identifier (Alibaba Cloud account ID) that is configured on the server and the custom ID of the machine group. If no user identifier or custom ID exists, no alert is triggered. If you have configured the settings, check whether the user identifier or custom ID is the same as the user identifier or custom ID in the returned message. If the user identifiers or custom IDs are different, use the following method to reconfigure the settings:
>
> - Configure a user identifier
> - Create a custom ID-based machine group

Run the `./checkingtool.sh --logFile [LogFileFullPath]` command to perform a quick check. If an Error message appears, fix the error as prompted.

> **Note** If the specified log file passes the check and the Logtail runtime environment is normal, we recommend that you view the error logs of the related parameters in the Log Service console. For more information, see How do I view Logtail collection errors?.



Common Logtail collection errors

You can use the automatic diagnostic tool of Logtail to identify the causes of errors that occur when Logtail collects logs and fix the errors based on the causes. The following table describes the causes of common Logtail collection errors and the related solutions.

Error	Solution
Installation files are missing.	Reinstall Logtail.
Logtail is not running.	Run the `/etc/init.d/ilogtaild start` command to start Logtail.
Multiple Logtail processes exist.	1. Run the `/etc/init.d/ilogtaild stop` command to stop Logtail. 2. Run the `/etc/init.d/ilogtaild start` command to start Logtail.
Port 443 is disabled.	Configure the firewall to enable port 443.
The configuration server cannot be found.	Check whether Logtail is installed on a Linux server. If Logtail is not installed on the Linux server, rerun the installation command. For more information, see Install Logtail on a Linux server.
The user configuration does not exist.	Check whether the following operations are performed: 1. A Logtail configuration is created. 2. The server is included in a machine group. 3. The Logtail configuration is applied to the machine group.
The specified log file cannot be matched.	Check whether the Logtail configuration is valid.
The specified log file is matched by more than one Logtail configurations.	If the log file is matched by more than one Logtail configurations, Logtail randomly selects a Logtail configuration. We recommend that you use only one Logtail configuration to match the log file.

Common parameters of the diagnostic tool

Parameter	Description
`--help`	Views the help documentation.
`--logFile [LogFileFullPath]`	Checks whether Logtail collects logs from the `LogFileFullPath` path and checks the properties of the basic runtime environment of Logtail, such as the integrity of installation files, runtime status, Alibaba Cloud account ID, and network connectivity.
`--logFileOnly [LogFileFullPath]`	Checks whether Logtail collects logs from the `LogFileFullPath` path.
`--envOnly`	Checks the properties of the runtime environment of Logtail.

9.11. How do I update a Logtail configuration after I switch the network type of an ECS instance from the classic network to a VPC?

If you switch the network type of an Elastic Compute Service (ECS) instance from the classic network to a virtual private cloud (VPC), you must restart Logtail and update the configurations of the related machine group. This way, Logtail can continue to collect logs from the ECS instance.

Procedure

1. Restart Logtail as an administrator.
   - Linux

     ```
     sudo /etc/init.d/ilogtaild stop
     sudo /etc/init.d/ilogtaild start
     ```

   - Windows
     a. Open the Run command window and enter `services.msc` to open the Services window.
     b. Restart the LogtailWorker service.
2. Update the configurations of the machine group.
   - IP address-based machine group

     If your machine group is an IP address-based machine group, replace the IP address in the machine group with the IP address that is obtained after you restart Logtail. You can obtain the IP address from the ip field in the *app_info.json* file. The file path of *app_info.json* varies based on the operating system.

     - Linux: */usr/local/ilogtail/app_info.json*
     - 64-bit Windows: *C:\Program Files (x86)\Alibaba\Logtail\app_info.json*
     - 32-bit Windows: *C:\Program Files\Alibaba\Logtail\app_info.json*

   - Custom ID-based machine group

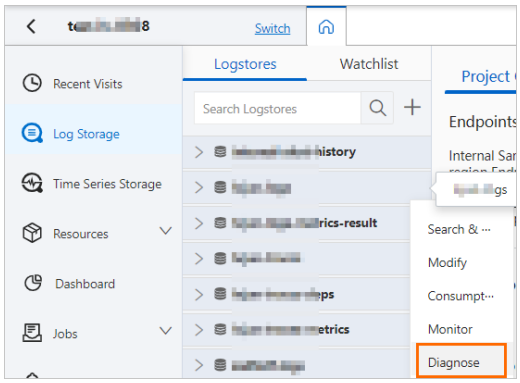
     If your machine group is a custom ID-based machine group, you do not need to update the configurations of the machine group.

9.12. How do I view Logtail collection errors?

When you use Logtail to collect logs, errors may occur. For example, regular expressions may fail to be parsed, invalid file paths may exist, and traffic may exceed the processing capabilities of shards. Log Service provides the automatic diagnostic tool to help you diagnose the errors that occur when Logtail collects logs.

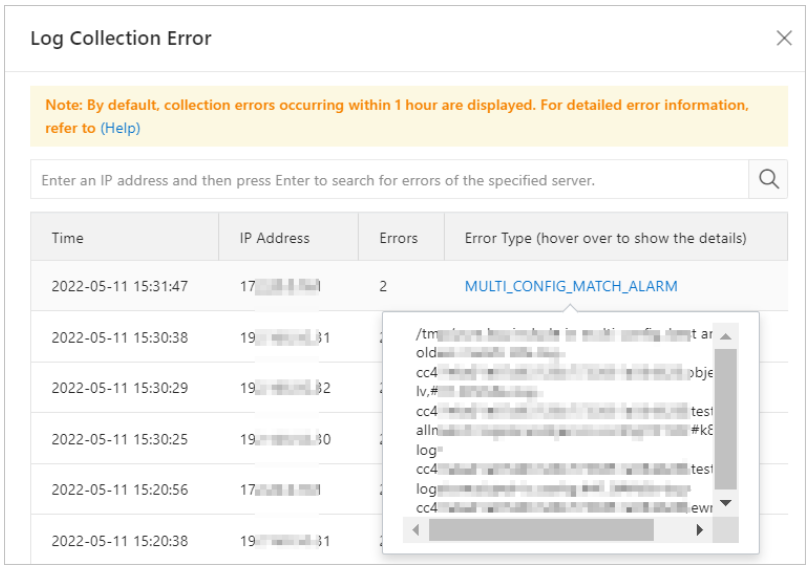
Procedure

1.
2.
3. On the **Logstores** tab, click the icon next to the Logstore that you want to manage, and then click **Diagnose**.



4. View log collection errors.

   The **Log Collection Error** panel displays all Logtail collection errors of the Logstore. You can click an error code to view the details of the error. For more information, see How do I troubleshoot the common errors that occur when Log Service collects logs?.



5. Query the collection errors that occurred on a server.

   In the **Log Collection Error** panel, enter the IP address of the server to view all collection errors that occurred on the server.

   After you fix all existing errors, you can check whether the errors persist. Historical errors are displayed before they expire. You can ignore these errors and check only the errors that are reported after you fix the historical errors. Logtail reports errors every 10 minutes.

> **Note** To view the complete logs that are dropped due to parsing failures, log on to the server and check the */usr/local/ilogtail/ilogtail.LOG* file.

9.13. How do I troubleshoot the common errors that occur when Log Service collects logs?

This topic describes the common errors that occur when Log Service collects data and the solutions to the errors.

If you encounter the errors that are not described in this topic, submit a ticket.

Error	Description	Solution
LOG_GROUP_WAIT_TOO_LONG_ALARM	After a data packet is generated, the system waits a long time to send the packet.	Check whether the system sends packets as expected, the data volume exceeds the default limit, the quota is insufficient, or network errors occur.
LOGFILE_PERMINSSION_ALARM	Logtail has no permissions to read the specified file.	Check whether the startup account of Logtail on the server is root. The root account is recommended.
SPLIT_LOG_FAIL_ALARM	Logtail fails to split logs into lines because the regular expression that is specified to match the beginning of the first line of a log does not match the content in the logs.	Check whether the regular expression is correct. If you want to collect single-line logs, you can specify `.*` as the regular expression.
MULTI_CONFIG_MATCH_ALARM	By default, you can use only one Logtail configuration to collect logs from a log file. If you use multiple Logtail configurations to collect logs from a log file, only one Logtail configuration takes effect. **Note** You can use multiple Logtail configurations to collect the stdout and stderr of Docker containers.	• Delete redundant Logtail configurations. • Modify configurations to allow Logtail to collect logs from a log file by using multiple Logtail configurations. For more information, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?.
REGEX_MATCH_ALARM	In full regex mode, log content does not match the regular expression that is specified.	Copy the sample log in the error details to generate a new regular expression.
PARSE_LOG_FAIL_ALARM	In modes such as JSON and delimiter, Logtail fails to parse logs because the log format does not conform to the defined format.	Click the error link to view details.

Error	Description	Solution
CATEGORY_CONFIG_ALARM	The Logtail configuration is invalid.	A common reason is that the specified regular expression fails to extract a part of the file path as a topic. If this error is caused by a different reason, submit a ticket.
LOGTAIL_CRASH_ALARM	Logtail stops responding because its server resource usage exceeds the upper limit.	Change the upper limits of CPU utilization and memory usage to larger values. For more information, see Configure the startup parameters of Logtail.
REGISTER_INOTIFY_FAIL_ALARM	Logtail fails to register the log listener in Linux. This error may occur because Logtail has no permissions to access the folder that Logtail listens on or the folder has been deleted.	Check whether Logtail has the permissions to access the folder or the folder is deleted.
DISCARD_DATA_ALARM	The CPU resources configured for Logtail are insufficient, or throttling is triggered on sending network data.	Change the upper limit of CPU utilization or the limit of concurrent operations to send network data to a larger value. For more information, see Configure the startup parameters of Logtail.
SEND_DATA_FAIL_ALARM	• No AccessKey pair is created for your Alibaba Cloud account. • The server on which Logtail runs cannot connect to Log Service, or network quality is poor. • The write quota is insufficient on Log Service.	• Create an AccessKey pair for your Alibaba Cloud account. • Check the local configuration file */usr/local/ilogtail/ilogtail_config.json* and run the **curl <Server address>** command to check whether any content is returned. • Increase the number of shards for the Logstore so that more data can be written to the Logstore.
REGISTER_INOTIFY_FAIL_ALARM	Logtail fails to register the inotify watcher for the log directory.	Check whether the log directory exists and check the permission settings of the directory.
SEND_QUOTA_EXCEED_ALARM	The log write traffic exceeds the limit.	Increase the number of shards in the Log Service console. For more information, see Split a shard.
READ_LOG_DELAY_ALARM	Log collection lags behind log generation. In most cases, this error occurs because the CPU resources configured for Logtail are insufficient or throttling is triggered on sending network data.	Change the upper limit of CPU utilization or the limit of concurrent operations to send network data to a larger value. For more information, see Configure the startup parameters of Logtail.

Error	Description	Solution
DROP_LOG_ALARM	Log collection lags behind log generation, and the number of log files that are generated during rotation and are not parsed exceeds 20. In most cases, this error occurs because the CPU resources configured for Logtail are insufficient or throttling is triggered on sending network data.	Change the upper limit of CPU utilization or the limit of concurrent operations to send network data to a larger value. For more information, see Configure the startup parameters of Logtail.
LOGDIR_PERMINSSION_ALARM	Logtail has no permissions to read the log directory.	Check whether the log directory exists. If the directory exists, check the permission settings of the directory.
ENCODING_CONVERT_ALARM	Encoding fails.	Check whether the configuration for log encoding is consistent with the actual implementation of log encoding.
OUTDATED_LOG_ALARM	The logs are expired. The log time lags behind the collection time for more than 12 hours. Possible causes: • The log parsing progress lags behind the expected time for more than 12 hours. • The custom time field is incorrectly configured. • The time output of the log recording program is invalid.	• Check whether the READ_LOG_DELAY_ALARM error is reported. If the error is reported, fix the error. If the error is not reported, check the configuration of the time field. • Check the configuration of the time field. If the time field is correctly configured, check whether the time output of the log recording program is valid.
STAT_LIMIT_ALARM	The number of files in the log directory that is specified in the Logtail configuration exceeds the limit.	Check whether the log directory contains a large number of files and subdirectories. Reconfigure the log directory for monitoring and the maximum number of levels of subdirectories that you want to monitor. You can also modify the mem_usage_limit parameter. For more information, see Configure the startup parameters of Logtail.

Error	Description	Solution
DROP_DATA_ALARM	When the Logtail process exits, logs are dumped to the local disk. However, the dump operation times out. As a result, the logs that are not dumped to the local disk are discarded.	In most cases, this error occurs because collection is severely blocked. You can change the upper limit of CPU utilization or the limit of concurrent operations to send network data to a larger value. For more information, see Configure the startup parameters of Logtail.
INPUT_COLLECT_ALARM	An error occurs when data is collected from the input data source.	Fix the error based on the error details.
HTTP_LOAD_ADDRESS_ALARM	The value of Addresses specified in the Logtail configuration that is used to collect HTTP data is invalid.	Specify a valid value for Addresses.
HTTP_COLLECT_ALARM	An error occurs in collecting HTTP data.	Fix the error based on the error details. In most cases, this error is caused by timeout.
FILTER_INIT_ALARM	An error occurs in initializing the filter.	In most cases, this error is caused by the invalid regular expressions of the filter. Fix the error based on the error details.
INPUT_CANAL_ALARM	An error occurs in the plug-in that is used to collect MySQL binary logs.	Fix the error based on the error details. When a Logtail configuration is updated, the canal service may restart. If the error is caused by the service restart, you can ignore the error.
CANAL_INVALID_ALARM	The plug-in that is used to collect MySQL binary logs is abnormal.	In most cases, this error is caused by inconsistent metadata. Metadata inconsistency may occur due to table scheme changes during running. Check whether changes are made to table schemas in the period during which the error is repeatedly reported. If the error is caused by a different reason, submit a ticket.
MYSQL_INIT_ALARM	An error occurs during MySQL initialization.	Fix the error based on the error details.
MYSQL_CHECKPOING_ALARM	The format of the checkpoints that are used for MySQL data collection is invalid.	Check whether to modify the checkpoint-related settings in the Logtail configuration. If the error is caused by a different reason, submit a ticket.

Error	Description	Solution
MYSQL_TIMEOUT_ALARM	The MySQL query times out.	Check whether the MySQL server is properly connected to the network.
MYSQL_PARSE_ALARM	The MySQL query results fail to be parsed.	Check whether the format of the checkpoints that are used for MySQL data collection matches the format of the required fields.
AGGREGATOR_ADD_ALARM	The system fails to add data to the queue.	Data is sent too fast. If large amounts of data need to be sent, you can ignore this error.
ANCHOR_FIND_ALARM	An error occurs in the processor_anchor plug-in, an error occurs in the Logtail configuration, or logs that do not match the Logtail configuration exist.	Click the error link to view the sub-type of the error. The following sub-types are available. You can check settings based on the error details of each sub-type. • `anchor cannot find key`: SourceKey is configured in the Logtail configuration, but the specified fields are not found in logs. • `anchor no start`: The system cannot find a match for Start in the value of SourceKey. • `anchor no stop`: The system cannot find a match for Stop in the value of SourceKey.
ANCHOR_JSON_ALARM	An error occurs in the processor_anchor plug-in. The plug-in fails to expand the JSON data that is extracted based on the configured Start and Stop.	Click the error link to view details. Examine collected logs and related configurations to check whether configuration errors or invalid logs exist.
CANAL_RUNTIME_ALARM	An error occurs in the plug-in that is used to collect MySQL binary logs.	Click the error link to view details and perform troubleshooting based on the details. In most cases, this error is related to the primary ApsaraDB RDS for MySQL instance that is connected.
CHECKPOINT_INVALID_ALARM	The system fails to parse checkpoints.	Click the error link to view details and perform troubleshooting based on the details and the key-value pairs of the checkpoints in the details. The values of the checkpoints are indicated by the first 1,024 bytes in the checkpoint file.

Error	Description	Solution
DIR_EXCEED_LIMIT_ALARM	The number of directories that Logtail listens on at the same time exceeds the limit.	Check whether the Logtail configurations whose data is stored to the current Logstore and other Logtail configurations on the server on which Logtail is installed involve a large number of subdirectories. Reconfigure the log directory for monitoring and the maximum number of levels of subdirectories that you want to monitor for each Logtail configuration.
DOCKER_FILE_MAPPING_ALARM	The system fails to add a Docker file mapping by running a Logtail command.	Click the error link to view details and perform troubleshooting based on the details and the command in the details.
DOCKER_FILE_MATCH_ALARM	The specified file cannot be found in the Docker container.	Click the error link to view details and perform troubleshooting based on the container information and the file path that is used for the search.
DOCKER_REGEX_COMPILE_ALARM	An error occurs in the service_docker_stdout plug-in. Compiling based on BeginLineRegex fails.	Click the error link to view details and check whether the regular expression in the details is correct.
DOCKER_STDOUT_INIT_ALARM	The service_docker_stdout plug-in fails to be initialized.	Click the error link to view the sub-type of the error. The following sub-types are available: • `host...version...error`: You can check whether the Docker engine specified in the Logtail configuration is accessible. • `load checkpoint error`: The system fails to load the checkpoint file. If this error does not affect your business, you can ignore this error. • `container...`: The specified container has invalid label values. Only stdout and stderr are supported. Perform troubleshooting based on the error details.
DOCKER_STDOUT_START_ALARM	The stdout size exceeds the limit when the service_docker_stdout plug-in is used to collect data.	In most cases, this error occurs because the stdout already exists when you use the plug-in for the first time. You can ignore this error.

Error	Description	Solution
DOCKER_STDOUT_STAT_ALARM	The service_docker_stdout plug-in cannot find the stdout.	In most cases, this error occurs because no stdout is available when a container terminates. You can ignore this error.
FILE_READER_EXCEED_ALARM	The number of files that Logtail opens at the same time exceeds the limit.	In most cases, this error occurs because Logtail is collecting logs from a large number of files. Check whether the settings of the Logtail configuration are proper.
GEOIP_ALARM	An error occurs in the processor_geoip plug-in.	Click the error link to view the sub-type of the error. The following sub-types are available: • `invalid ip...` : The system fails to obtain an IP address. Check whether SourceKey in the Logtail configuration is correctly configured or whether invalid logs exist. • `parse ip...` : The system fails to parse an IP address into a city. Perform troubleshooting based on the error details. • `cannot find key...` : The system cannot find a match for SourceKey in logs. Check whether the Logtail configuration is correct or whether invalid logs exist.
HTTP_INIT_ALARM	An error occurs in the metric_http plug-in. The plug-in fails to compile the regular expression specified by the ResponseStringMatch parameter in the Logtail configuration.	Click the error link to view details and check whether the regular expression in the details is correct.
HTTP_PARSE_ALARM	An error occurs in the metric_http plug-in. The plug-in fails to obtain HTTP responses.	Click the error link to view details and check the Logtail configuration or the requested HTTP server based on the details.
INIT_CHECKPOINT_ALARM	An error occurs in the plug-in that is used to collect binary logs. The plug-in fails to load the checkpoint file and starts data processing from the beginning without a checkpoint.	Click the error link to view details and determine whether to ignore the error based on the details.

Error	Description	Solution
LOAD_LOCAL_EVENT_ALARM	Logtail performs local event handling.	This error does not usually occur. If this error is caused by a non-human operation, you must perform troubleshooting. You can click the error link to view details and perform troubleshooting based on the file name, Logtail configuration name, project, and Logstore in the details.
LOG_REGEX_FIND_ALARM	Errors occur in the processor_split_log_regex and processor_split_log_string plug-ins. The plug-ins cannot find a match for SplitKey in logs.	Click the error link to view details and check whether configuration errors exist.
LUMBER_CONNECTION_ALARM	An error occurs in the service_lumberjack plug-in. The server is shut down while the plug-in is stopped.	Click the error link to view details and perform troubleshooting based on the details. In most cases, you can ignore this error.
LUMBER_LISTEN_ALARM	An error occurs in the service_lumberjack plug-in. The plug-in fails to perform listening during initialization.	Click the error link to view the sub-type of the error. The following sub-types are available: • `init tls error...` : Check whether TLS-related configurations are correct based on the error details. • `listen init error...` : Check whether address-related configurations are correct based on the error details.
LZ4_COMPRESS_FAIL_ALARM	An error occurs when Logtail performs LZ4 compression.	Click the error link to view details and perform troubleshooting based on the values of log lines, project, category, and region in the details.

Error	Description	Solution
MYSQL_CHECKPOINT_ALARM	An error occurs in the plug-in that is used for MySQL data collection. The error is related to checkpoints.	Click the error link to view the sub-type of the error. The following sub-types are available: • `init checkpoint error...`: Checkpoint initialization fails. Check the checkpoint column specified by the Logtail configuration based on the error details and check whether the obtained values are correct. • `not matched checkpoint...`: Logs do not match the checkpoint information. Check whether the mismatch is caused by user operations such as configuration update based on the error details. If the mismatch is caused by user operations, you can ignore the error.
NGINX_STATUS_COLLECT_ALARM	An error occurs in the nginx_status plug-in. The plug-in fails to obtain status information.	Click the error link to view details and perform troubleshooting based on the details and the URLs in the details.
NGINX_STATUS_INIT_ALARM	An error occurs in the nginx_status plug-in. The plug-in fails to initialize the URLs specified in parsing configurations.	Click the error link to view details and check whether the URLs in the details are correct.
OPEN_FILE_LIMIT_ALARM	Logtail fails to open the file because the number of opened files exceeds the limit.	Click the error link to view details and perform troubleshooting based on the file path, project, and Logstore in the details.
OPEN_LOGFILE_FAIL_ALARM	An error occurs when Logtail opens the file.	Click the error link to view details and perform troubleshooting based on the file path, project, and Logstore in the details.

Error	Description	Solution
PARSE_DOCKER_LINE_ALARM	An error occurs in the service_docker_stdout plug-in. The plug-in fails to parse the log.	Click the error link to view the sub-type of the error. The following sub-types are available: • `parse docker line error: empty line` : The log is empty. • `parse json docker line error ...` : The system fails to parse the log into the JSON format. Perform troubleshooting based on the error details and the first 512 bytes of the log. • `parse cri docker line error. ..` : The system fails to parse the log into the CRI format. Perform troubleshooting based on the error details and the first 512 bytes of the log.
PLUGIN_ALARM	An error occurs in initializing and calling plug-ins.	Click the error link to view the sub-type of the error. The following sub-types are available. You can perform troubleshooting based on the error details. • `init plugin error...` : The system fails to initialize a plug-in. • `hold on error...` : The system fails to suspend a plug-in. • `resume error...` : The system fails to resume a plug-in. • `start service error...` : The system fails to start a plug-in of the service input type. • `stop service error...` : The system fails to stop a plug-in of the service input type.
PROCESSOR_INIT_ALARM	An error occurs in the processor_regex plug-in. The plug-in fails to compile the regular expression specified in the Logtail configuration.	Click the error link to view details and check whether the regular expression in the details is correct.

Error	Description	Solution
PROCESS_TOO_SLOW_ALARM	Logtail parses logs too slowly.	1. Click the error link to view details and check whether the parsing speed is acceptable based on the log quantity, buffer size, and parsing time in the details. 2. If the speed is too slow, check whether other processes on the server on which Logtail is installed occupy excessive CPU resources or whether inappropriate parsing configurations exist, such as inefficient regular expressions.
REDIS_PARSE_ADDRESS_ALARM	An error occurs in the redis plug-in. The plug-in fails to parse the value of ServerUrls provided in the Logtail configuration.	Click the error link to view details. Check the URLs for which the error is reported.
REGEX_FIND_ALARM	An error occurs in the processor_regex plug-in. The plug-in fails to find the fields specified by SourceKey in logs.	Click the error link to view details. Check whether SourceKey is correctly configured or whether the logs are valid.
REGEX_UNMATCHED_ALARM	An error occurs in the processor_regex plug-in. The match operation of the plug-in fails.	Click the error link to view the sub-type of the error. The following sub-types are available. You can perform troubleshooting based on the error details. • `unmatch this log content...`: The system fails to match logs against the regular expression specified in the Logtail configuration. • `match result count less...`: The number of matched fields is less than the number of fields specified by Keys in the Logtail configuration.
SAME_CONFIG_ALARM	Duplicate Logtail configurations are found for a Logstore. The Logtail configuration that is most recently found is discarded.	Click the error link to view details. Check whether configuration errors exist based on the details and the Logtail configuration path in the details.
SPLIT_FIND_ALARM	Errors occur in the split_char and split_string plug-ins. The plug-ins fail to find the fields specified by SourceKey in logs.	Click the error link to view details. Check whether SourceKey is correctly configured or whether the logs are valid.

Error	Description	Solution
SPLIT_LOG_ALARM	Errors occur in the processor_split_char and processor_split_string plug-ins. The number of parsed fields is different from the number of fields specified by SplitKeys.	Click the error link to view details. Check whether SourceKey is correctly configured or whether the logs are valid.
STAT_FILE_ALARM	An error occurs when the LogFileReader object is used to collect data from a file.	Click the error link to view details and perform troubleshooting based on the details and the file path in the details.
SERVICE_SYSLOG_INIT_ALARM	An error occurs in the service_syslog plug-in. The plug-in fails to be initialized.	Click the error link to view details. Check whether Address in the Logtail configuration is correctly configured.
SERVICE_SYSLOG_STREAM_ALARM	An error occurs in the service_syslog plug-in. The plug-in fails to collect data over TCP.	Click the error link to view the sub-type of the error. The following sub-types are available. You can perform troubleshooting based on the error details. • `accept error...` : An error occurs when the Accept command is run. The plug-in waits for a while and try again. • `setKeepAlive error...` : The system fails to configure Keep Alive. The plug-in skips this error and continues. • `connection i/o timeout...` : A TCP read times out. The plug-in modifies the timeout period and continues to read data. • `scan error...` : A TCP read error occurs. The plug-in waits for a while and try again.
SERVICE_SYSLOG_PACKET_ALARM	An error occurs in the service_syslog plug-in. The plug-in fails to collect data over UDP.	Click the error link to view the sub-type of the error. The following sub-types are available. You can perform troubleshooting based on the error details. • `connection i/o timeout...` : A UDP read times out. The plug-in modifies the timeout period and continues to read data. • `read from error...` : A UDP read error occurs. The plug-in waits for a while and try again.

Error	Description	Solution
PARSE_TIME_FAIL_ALARM	The system fails to parse the log time.	You can use one of the following methods to identify the cause of the error and fix the error: • Check whether the time field extracted by using a regular expression is correct. • Check whether the value of the specified time field matches the time expression specified in the Logtail configuration.

9.14. How do I optimize regular expressions?

You can optimize regular expressions to improve the Logtail collection performance.

The following describes some suggestions about how to optimize regular expressions:

- Use precise characters.

  Do not arbitrarily use `.*` to match fields because this regular expression can match with a wide range of search results. Such actions can to lead to mismatches occurring or a decrease in matching performance. For example, to return results of fields that consist only of letters, use `[A-Za-z]`.

- Use correct measure words.

  Do not arbitrarily use plus signs (+), commas (,), or asterisks. For example, to match target IP addresses, use `\d` instead of `\d+` or `\d{1,3}` because of its higher efficiency.

- Debug multiple times.

  Debugging is similar to troubleshooting. You can debug the time consumed by your regular expressions at the **Regex101** website, and promptly optimize them if there is a large amount of backtracking.

9.15. How do I collect different types of logs in full regex mode?

If you want to collect logs in full regex mode, the logs that you want to collect must be of the same type. If the logs that you want to collect are of different types, you can use the schema-on-read and schema-on-write approaches to process the logs before you collect the logs in full regex mode.

Java logs are program logs that contain normal information and errors such as stack exceptions. Java logs can be one of the following logs:

- Multi-line WARNING logs
- Single-line INFO logs
- Key-value DEBUG logs

```
[2018-10-01T10:30:31,000] [WARNING] java.lang.Exception: another exception happened
    at TestPrintStackTrace.f(TestPrintStackTrace.java:3)
    at TestPrintStackTrace.g(TestPrintStackTrace.java:7)
    at TestPrintStackTrace.main(TestPrintStackTrace.java:16)
[2018-10-01T10:30:32,000] [INFO] info something
[2018-10-01T10:30:33,000] [DEBUG] key:value key2:value2
```

You can use the following approaches to process logs before you collect the logs:

- You can use the schema-on-write approach and specify different regular expressions for multiple Logtail configurations. This way, you can use multiple Logtail configurations to collect logs from a log file and extract the fields that you specify.

  > **Note** By default, you can use only one Logtail configuration to collect logs from a log file. For more information about how to use multiple Logtail configurations to collect logs from a log file, see What do I do if I want to use multiple Logtail configurations to collect logs from a log file?.

- You can use the schema-on-read approach and specify a regular expression that contains the common fields of logs to collect the logs.

  For example, if you want to collect multi-line logs, you can specify a regular expression in which the timestamp and level of the logs are specified to match the beginning of the first line of a log, and the rest of the log is included in the message field. If you want to analyze content in the message field, you can create an index for the message field, specify a regular expression to extract the content that you want to analyze from the message field, and then analyze the content.

  > **Note** We recommend that you use this approach only for scenarios in which you need to analyze tens of millions of logs or less.

9.16. Why am I unable to collect SLB access logs?

This topic describes how to troubleshoot in cases where you are unable to collect SLB access logs.

1. Check whether the access log collection function has been activated for SLB instances.

Activate the access log collection function for each SLB instance separately. Then, the generated access logs can be written into your Logstore in real time.

To do so, log on to the SLB console, and choose **Logs > Access Logs** to view the **Access Logs (Layer-7)** list.

- Verify that the specified SLB instance exists.
- Confirm the **Storage Path** of the SLB instance.

  This column displays information about the project and Logstore. In this case, make sure that you check whether SLB logs exist in the correct location in the console.

2. Check whether RAM users are correctly authorized.

During activation of the access log collection function, the system guides you through RAM user authorization. The function can be successfully activated only after RAM users are successfully authorized. If RAM users are incorrectly created or deleted, the collected logs cannot be delivered to your Logstore.

Troubleshooting

Log on to the RAM Console. On the **RAM Roles** page, check whether the AliyunLogArchiveRole role exists.

- **If AliyunLogArchiveRole does not exist**, use your Alibaba Cloud account to log on to the RAM console and click the quick authorization link to create the RAM users required for authorization.
- **If AliyunLogArchiveRole exists**, click the role name and check whether the role is correctly authorized.

  The following shows the default policy. If your policy has been modified, we recommend that you replace the current policy with the default policy.

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": [
        "log:PostLogStoreLogs"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

3. Check whether any log is generated.

If you do not find any SLB access log in the Log Service console, it is likely that no log has been generated. Possible causes include:

- Layer-7 listening is not configured for the current instance.

  Currently, only instances configured with layer-7 listening are supported. Common layer-7 listening protocols include HTTP and HTTPS. For more information, see Listener overview.

- Historical logs that were generated before activation of the access log collection function are not collected.

  Instead, only logs that are generated after activation of the access log collection function are collected.

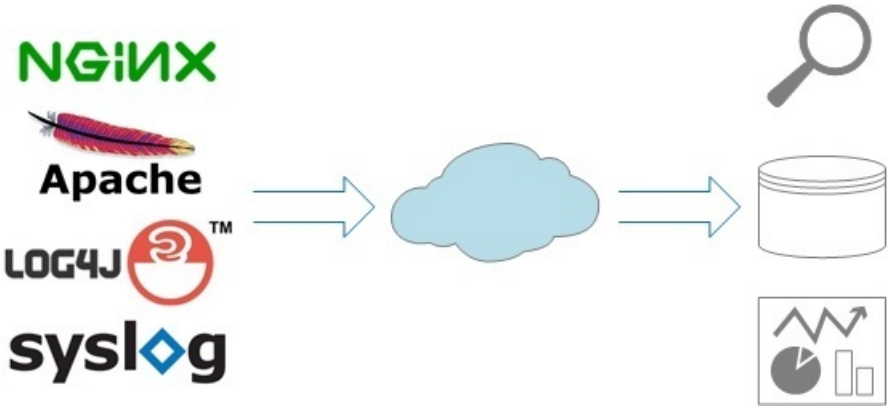
- The specified instance did not receive a request.

  Logs are generated only when you request access to the listener of the instance.

9.17. What are the differences among log collection agents?

Client evaluation in log collection scenarios

In the data technology (DT) era, hundreds of millions of servers, mobile terminals, and network devices generate a large number of logs every day. The centralized log processing solution effectively meets the log consumption requirements in the lifecycle of log data. Before consuming logs, you need to collect logs from devices and synchronize them to the cloud first.



Three log collection tools

- Logstash
  - As a part of the ELK Stack, Logstash is active in the open-source community. It can work with extensive plug-ins in the ecosystem.
  - Logstash is coded in JRuby and can run across platforms on Java virtual machines (JVMs).
  - With a modular design, Logstash features high scalability and interoperability.
- Fluentd
  - Fluendtd is a popular log collection tool in the open-source community. Its core component, td-agent, is commercially available and maintained by Treasure Data. Fluentd is selected for evaluation in this topic.
  - Fluentd is coded in CRuby. Some key components related to its performance are re-coded in C. The overall performance of Fluentd is excellent.
  - Fluentd features a simple design and provides reliable data transmission in pipelines.
  - Compared with Logstash, Fluentd has fewer plug-ins.
- Logtail
  - As the producer of Alibaba Cloud Log Service, Logtail has been tested in big data scenarios for many years in Alibaba Group.
  - Logtail, which is coded in C++, delivers excellent performance in stability, resource control, and management.
  - Compared with Logstash and Fluentd, Logtail obtains less support from the open-source community and focuses more on log collection.

Feature comparison

Feature	Logstash	Fluentd	Logtail
Log data read	Polling	Polling	Triggered by event
File rotation	Supported	Supported	Supported

Feature	Logstash	Fluentd	Logtail
Failover processing based on local checkpoints	Supported	Supported	Supported
General log parsing	Parsing by using Grok based on regular expressions	Parsing by using regular expressions	Parsing by using regular expressions
Specific log types	Mainstream formats such as delimiter, key-value, and JSON	Mainstream formats such as delimiter, key-value, and JSON	Mainstream formats such as delimiter, key-value, and JSON
Data compression for transmission	Supported by plug-ins	Supported by plug-ins	LZ4
Data filtering	Supported	Supported	Supported
Data buffer for transmission	Supported by plug-ins	Supported by plug-ins	Supported
Transmission exception handling	Supported by plug-ins	Supported by plug-ins	Supported
Runtime environment	Coded in JRuby and dependent on the JVM environment	Coded in CRuby and C and dependent on the Ruby environment	Coded in C++, without special requirements for the runtime environment
Thread support	Multithreading	Multithreading restricted by the global interpreter lock (GIL)	Multithreading
Hot upgrade	Not supported	Not supported	Supported
Centralized configuration management	Not supported	Not supported	Supported
Running status self-check	Not supported	Not supported	CPU or memory threshold protection supported

Performance comparison in log collection scenarios

For example, the following Nginx access log contains 365 bytes, from which 14 fields can be extracted:

In the simulated test scenario, this log is repeatedly written at different compression ratios. The time field of each log is set to the current system time when the log is written, and the other 13 fields are the same. Compared with the actual scenario, the simulated scenario has no difference in parsing logs. The only difference lies in that a high data compression ratio can reduce the network traffic on writing data.

Logstash

In Logstash 2.0.0, Logstash parses logs by using Grok and writes the logs to Kafka by using a built-in plug-in that enables GZIP compression.

Log parsing configuration:

```
grok {
    patterns_dir=>"/home/admin/workspace/survey/logstash/patterns"
    match=>{ "message"=>"%{IPORHOST:ip} %{USERNAME:rt} - \[%{HTTPDATE:time}\] \"%{WORD:meth
od} %{DATA:url}\" %{NUMBER:status} %{NUMBER:size} \"%{DATA:ref}\" \"%{DATA:agent}\" \"%{DAT
A:cookie_unb}\" \"%{DATA:cookie_cookie2}\" \"%{DATA:monitor_traceid}\" %{WORD:cell} %{WORD:
ups} %{BASE10NUM:remote_port}" }
    remove_field=>["message"]
}
```

The following table lists test results.

Write transactions per second (TPS)	Write traffic (Unit: KB/s)	CPU usage (Unit: %)	Memory usage (Unit: MB)
500	178.22	22.4	427
1,000	356.45	46.6	431
5,000	1,782.23	221.1	440
10,000	3,564.45	483.7	450

Fluentd

In td-agent 2.2.1, Fluentd parses logs by using regular expressions and writes the logs to Kafka by using the third-party plug-in fluent-plugin-kafka that enables GZIP compression.

Log parsing configuration:

```
<source>
  type tail
  format /^(? <ip>\S+)\s(? <rt>\d+)\s-\s\[(? <time>[^\]]*)\]\s"(? <url>[^\"]+)"\s(? <status
>\d+)\s(? <size>\d+)\s"(? <ref>[^\"]+)"\s"(? <agent>[^\"]+)"\s"(? <cookie_unb>\d+)"\s"(? <c
ookie_cookie2>\w+)"\s"(?
<monitor_traceid>\w+)"\s(? <cell>\w+)\s(? <ups>\w+)\s(? <remote_port>\d+).*$/
  time_format %d/%b/%Y:%H:%M:%S %z
  path /home/admin/workspace/temp/mock_log/access.log
  pos_file /home/admin/workspace/temp/mock_log/nginx_access.pos
  tag nginx.access
</source>
```

The following table lists test results.

Write TPS	Write traffic (Unit: KB/s)	CPU usage (Unit: %)	Memory usage (Unit: MB)
500	178.22	13.5	61
1,000	356.45	23.4	61
5,000	1,782.23	94.3	103

Note Due to the restrictions of the GIL, a single process of Fluentd uses only one CPU core. You can install the multiprocess plug-in to use multiple processes for achieving a higher log throughput.

Logtail

In Logtail 0.9.4, Logtail uses regular expressions to extract log fields, compresses data by using the LZ4 compression algorithm, and then writes the data to Alibaba Cloud Log Service in compliance with HTTP. The batch_size parameter is set to 4000.

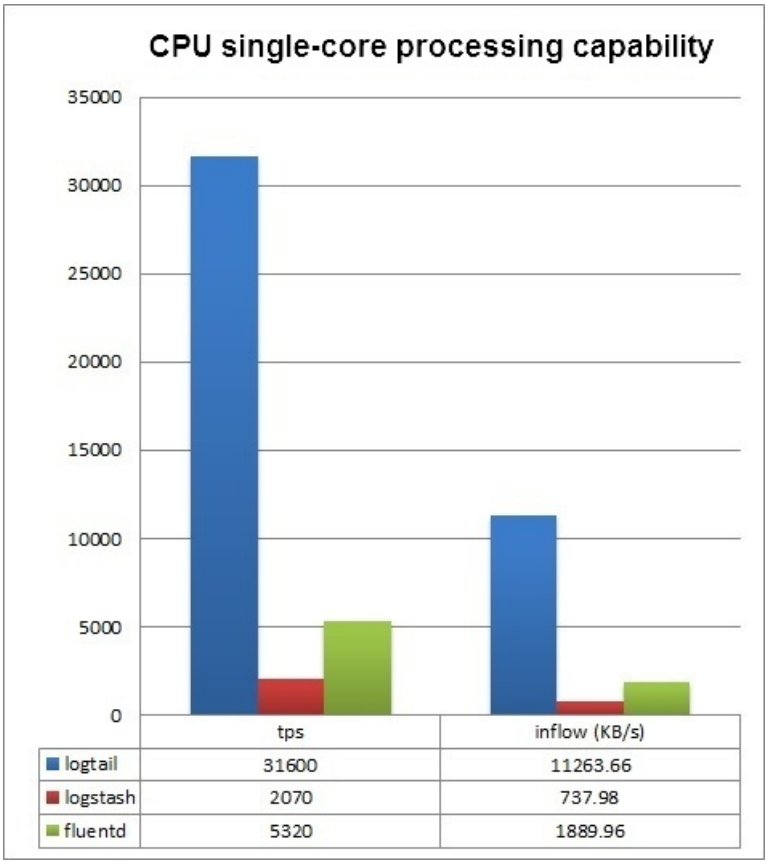
Log parsing configuration:

```
logRegex : (\S+)\s(\d+)\s-\s\[([^]]+)]\s"([^"]+)"\s(\d+)\s(\d+)\s"([^"]+)"\s"([^"]+)"\s"(\d
+)"\s"(\w+)"\s"(\w+)"\s(\w+)\s(\w+)\s(\d+).*
keys : ip,rt,time,url,status,size,ref,agent,cookie_unb,cookie_cookie2,monitor_traceid,cell,
ups,remote_port
timeformat : %d/%b/%Y:%H:%M:%S
```

The following table lists test results.

Write TPS	Write traffic (Unit: KB/s)	CPU usage (Unit: %)	Memory usage (Unit: MB)
500	178.22	1.7	13
1,000	356.45	3	15
5,000	1,782.23	15.3	23
10,000	3,564.45	31.6	25

Comparison of single-core CPU processing capabilities



Summary

The three log collection tools have their own advantages and disadvantages:

- Logstash supports all mainstream log types, the most abundant plug-ins, and flexible customization. However, its performance on log collection is relatively poor, and it requires high memory usage when running in the JVM environment.
- Fluentd supports all mainstream log types and many plug-ins. Its performance on log collection is excellent.
- Logtail occupies the fewest CPU and memory resources of machines, achieves a high performance throughput, and provides comprehensive support for common log collection scenarios. However, it lacks the support of plug-ins, so it is less flexible and scalable than the preceding two tools.

9.18. What are the differences between LogHub and Kafka?

Kafka is a distributed messaging system that features a high throughput and horizontal scalability. It is widely used for message publishing and subscription. Serving as open-source software, Kafka helps you build a Kafka cluster as required.

Log Service is a log platform service developed based on Apsara Distributed File System. It supports real-time collection, storage, distribution, and query of various types of logs. It provides external services by using the standard Restful API.

LogHub of Log Service provides public channels for log collection and distribution. You can use LogHub if you do not want to build or maintain a Kafka cluster.

Concept	Kafka	LogHub
Storage object	Topic	Logstore
Horizontal partitioning	Partition	Shard
Data consumption position	Offset	Cursor

Feature comparison

Feature	Kafka	LogHub
Dependency	On-premises or shared Kafka cluster	Log Service
Communications protocol	TCP	HTTP (Restful API) and port 80
Access control	None	Signature authentication and access control based on cloud accounts
Dynamic scaling	None	Auto scaling of shards, which can be dynamically merged or split without any impact on users
Multi-tenant QoS	None	Shard-based standard throttling
Number of data replicas	Customizable	Three replicas by default and not customizable
Failover or replication	Completed by using a tool	Automatically completed, which is imperceptible to users
Scaling or upgrade	Completed by using a tool, which affects services	Imperceptible to users
Write mode	Round robin or key hash	Round robin or key hash
Current consumption position	Stored in ZooKeeper of the Kafka cluster	Maintained on the server, which does not require your intervention
Data retention period	Specified in the configuration	Changed dynamically based on requirements

9.19. What do I do if I want to use multiple Logtail configurations to collect logs from a log file?

This topic describes how to use multiple Logtail configurations to collect logs from a log file.

By default, you can use only one Logtail configuration to collect logs from a log file. If you use multiple Logtail configurations to collect logs from a log file, only one Logtail configuration can be applied. If multiple Logtail configurations are applied, resources are consumed multiple times, and the performance of other services that are deployed on the same server as Logtail is affected. The resources include CPU, memory, disk I/O, and network I/O. We recommend that you do not use multiple Logtail configurations to collect logs from a log file.

If you want to store the same logs in different Logstores, you can use the data transformation feature of Log Service. The data transformation feature can replicate logs without affecting the performance of other services that are deployed on the same server as Logtail. For more information, see Replicate data from a Logstore.

If you want to use multiple Logtail configurations to collect logs from a log file, you can use one of the following methods:

- Create a symbolic link for a directory

  Create a symbolic link for the directory in which the log file is stored. For example, if you want to use two Logtail configurations to collect logs from the */home/log/nginx/log/log.log* file, you must run the following command to create a symbolic link that points to the directory of the file. Then, you can specify the real path in one Logtail configuration and specify the symbolic link in the other Logtail configuration.

  ```
  ln -s /home/log/nginx/log /home/log/nginx/link_log
  ```

- Add settings in Logtail configurations to forcefully collect logs

  If you want to use multiple Logtail configurations to forcefully collect logs from a log file, you can add `{"force_multiconfig": true}` in the **Advanced Options > More Configurations** section of the Logtail configurations.