Alibaba Cloud
Log Service

User Guide

Issue: 20180917

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 due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall 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 the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. 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 bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, 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 content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, 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 the Alibaba Cloud website, product programs, or content 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 contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

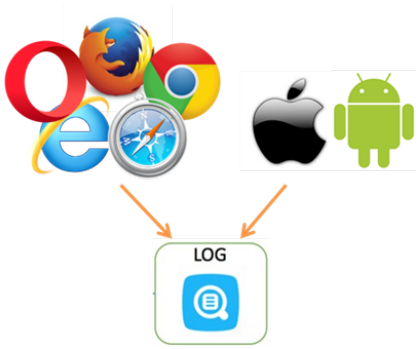
Style	Description	Example
	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 **Danger:** Resetting will result in the loss of user configuration data.
	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 **Warning:** Restarting will cause business interruption. About 10 minutes are required to restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 **Note:** Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 **Note:** You can use **Ctrl** + **A** to select all files.
>	Multi-level menu cascade.	**Settings** > **Network** > **Set network type**
Bold	It is used for buttons, menus, page names, and other UI elements.	Click **OK**.
`Courier font`	It is used for commands.	Run the `cd /d C:/windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich` *{stand | slave}*

Contents

1 Data Collection

1.1 Web Tracking

Log Service supports collecting logs from HTML, H5, iOS, and Android platforms by using Web Tracking, and customizing dimensions and metrics.



As shown in the preceding figure, you can collect user information from various browsers, iOS apps, and Android apps (apart from *iOS/Android SDK*) by using Web Tracking. For example:

- Browsers, operating systems, and resolutions used by users.
- Browsing behaviors of users, such as the clicking behaviors and purchasing behaviors on the website.
- The staying time in the app for users and whether the users are active or not.

Note:
Using Web Tracking means that this Logstore enables the anonymous write permission of the Internet, and dirty data may be generated.

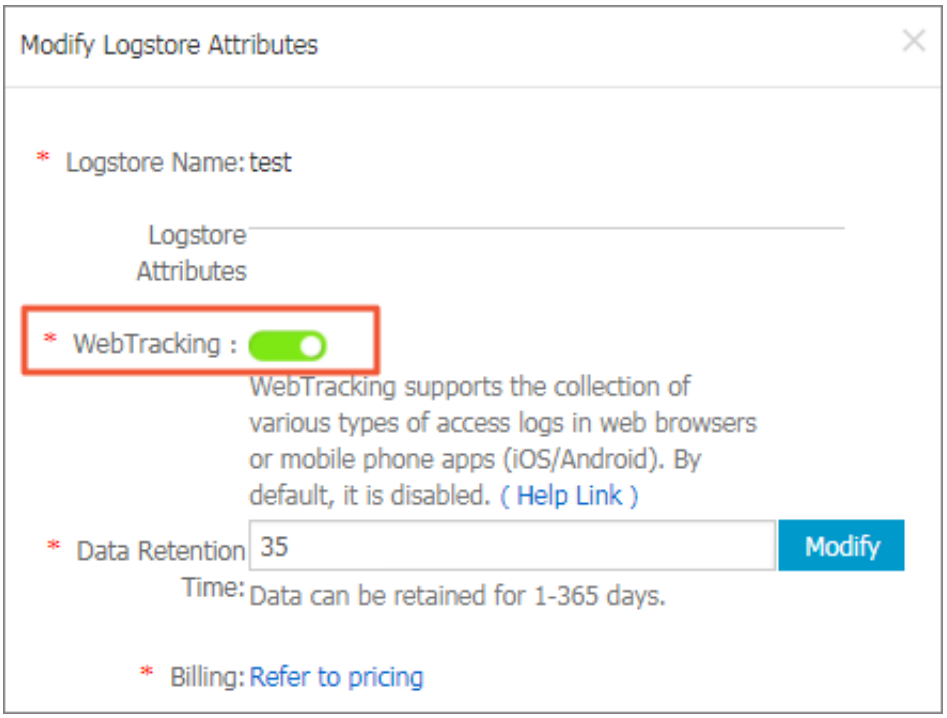
Procedure

Step 1. Enable Web Tracking

You can enable Web Tracking in the console or by using Java SDK.

- **Enable Web Tracking in the console**

  1. On the Logstore List page, click **Modify** at the right of the Logstore that must enable the Web Tracking function.
  2. Turn on the Web Tracking switch.



- **Enable Web Tracking by using**

  Java SDK:

```
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 the Web Tracking function on the created Logstore
.
          LogStore logSt = client.GetLogStore(project, logStore).
GetLogStore();
          client.UpdateLogStore(project, new LogStore(logStore,
logSt.GetTtl(), logSt.GetShardCount(), true));
          //Disable the Web Tracking function.
          //client.UpdateLogStore(project, new LogStore(logStore,
logSt.GetTtl(), logSt.GetShardCount(), false));
          //Create a Logstore that supports the Web Tracking
function.
          //client.UpdateLogStore(project, new LogStore(logStore, 1
, 1, true));
      }
      catch (LogException e){
          e.printStackTrace();
      }
  }
```

```
}
```

Step 2. Collect logs

After the Web Tracking function is enabled for Logstore, you can use any of the following three methods to upload data to the Logstore.

- **Use HTTP GET request**

```
curl --request GET 'http://${project}.${host}/logstores/${logstore}/
track? APIVersion=0.6.0&key1=val1&key2=val2'
```

The parameter meanings are as follows.

Field	Meaning
`${project}`	The name of the project created in Log Service.
`${host}`	The domain name of the region where your Log Service is located.
`${logstore}`	The name of the Logstore with the Web Tracking function enabled under `${project}` .
`APIVersion=0.6.0`	The reserved field, which is required.
`__topic__=yourtopic`	Specify the log topic, reserved fields (optional).
`key1=val1`, `key2=val2`	The key-value pairs to be uploaded to Log Service. Multiple key-value pairs are supported, but you must make sure that the URL length is less than 16 KB.

- **Use HTML img tag**

```
<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 parameter meanings are the same as those in Use HTTP GET request.

- **Use JS SDK**

1. Copy *loghub-tracking.js* to the *web* directory, and introduce the following script on the page:

Click to download.

```
<script type="text/javascript" src="loghub-tracking.js" async></
script>
```

Note:

To keep page loading running, the script sends HTTP requests asynchronously. If data must be sent several times in the page loading process, the subsequent request overwrites the preceding HTTP request, and the browser shows the tracking request exits. Sending requests synchronously can help to avoid this problem. To send requests synchronously, replace the statement in the script.

Original script:

```
this.httpRequest_.open("GET", url, true)
```

Replace the last parameter to send requests synchronously:

```
this.httpRequest_.open("GET", url, false)
```

2. Create a Tracker object.

```
var logger = new window.Tracker('${host}','${project}','${logstore
}');
logger.push('customer', 'zhangsan');
logger.push('product', 'iphone 6s');
logger.push('price', 5500);
logger.logger();
logger.push('customer', 'lisi');
logger.push('product', 'ipod');
logger.push('price', 3000);
logger.logger();
```

The parameter meaning are as follows:

Field	Meaning
`${host}`	The domain name of the region where your Log Service is located.
`${project}`	The name of the project created in Log Service.
`${logstore}`	The name of the Logstore with the Web Tracking function enabled under `${project}`.

After running the preceding commands, you can see the following two logs in Log Service:

```
customer:zhangsan
product:iphone 6s
price:5500
```

```
customer:lisi
product:ipod
price:3000
```

After data is uploaded to Log Service, you can use Log Service to *ship* data to Object Storage Service (OSS). You can also use the Consumer Library provided by Log Service to consume data.

1.2 Logstash

1.2.1 Quick installation

You can choose to install logtash quickly on your server by default.

Context

Log Service provides an installation package based on Logstash 2.2.2, which integrates with JRE 1.8, Log Service write plug-in, and NSSM 2.24. The deployment process by using this package is simpler than *Custom installation*. You can select the custom installation for complex requirements.

Procedure

1. Download and extract the *installation package* to the C: drive.
2. Confirm the Logstash startup program path is *C:\logstash-2.2.2-win\bin\logstash.bat*.

1.2.2 Custom installation

You can install Logstash by using quick installation or custom installation methods.

Context

When you have other requirements for logstroudsburg's installation configuration, you can choose how you want to customize the installation, modify the default installation configuration.

Procedure

1. Install Java
   1. Download the installation package.

      Go to the *Java official website* to download JDK for installation.

2. Sets the environment variable.

   Add or modify environment variables in advanced system settings.

   - **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*

3. Perform verification.

   Run `PowerShell` or `cmd.exe` for verification.

```
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

   1. Download the installation package from the official website.

      Select version 2.2 or later on the *Logstash* home page.

   2. Install Logstash.

      Extract *logstash-2.2.2.zip* to the *C:\logstash-2.2.2* directory.

      Confirm the Logstash startup program path is *C:\logstash-2.2.2\bin\logstash.bat* .

3. Install the plug-in used by Logstash to write logs to Log Service

   Install the plug-in online or offline based on the network environment where the machine resides.

   - Online installation

     The plug-in is hosted by RubyGems. For more information, see *here* .

     Run `PowerShell` or `cmd.exe` to go to the Logstash installation directory.

```
PS C:\logstash-2.2.2> .\bin\plugin install logstash-output-logservice
```

   - Offline installation

     Download from the official website. Go to the *logstash-output-logservice* page and click **Download** in the lower-right corner.

If the machine from which logs are collected cannot access the Internet, copy the downloaded gem package to the *C:\logstash-2.2.2* directory of the machine. Run *PowerShell* or *cmd.exe* to go to the Logstash installation directory. Perform the following command to install lLogstash:

```
PS C:\logstash-2.2.2> .\bin\plugin install C:\logstash-2.2.2\
logstash-output-logservice-0.2.0.gem
```

- Perform verification.

```
PS C:\logstash-2.2.2> .\bin\plugin list
```

Verify that logstash-output-logservice exists in the installed plug-in list of the machine.

4. Install NSSM

Download from the official website. Go to the *NSSM official website* to download the NSSM installation package.

fter you download the installation package to the local machine, extract it to the *C:\logstash-2.2.2\nssm-2.24*.

1.2.3 Set Logstash as a Windows service

When logstash.bat is started in PowerShell, the Logstash process is working in the frontend. Logstash is generally used for testing configurations and debugging collections. Therefore, we recommend that you set Logstash as a Windows service after the debugging is passed so as to enable Logstash to work in the backend and start automatically when power-on.

Besides setting Logstash as a Windows service, you can also start, stop, modify, and delete the service by using command lines. For more information about how to use NSSM, see *NSSM official document*.

Add Logstash as a Windows service

This operation is generally performed when Logstash is deployed for the first time. If Logstash has been added, skip this step.

Run the following command to add 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

If the configuration file in the Logstash *conf* directory is updated, stop the Logstash service and then start it again.

Run the following command 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

Run the following command 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

Run the following command 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

Run the following command 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
```

1.2.4 Create Logstash collection configurations

Context

Related plug-ins

- **logstash-input-file**

  This plug-in is used to collect log files in tail mode. For more information, see*logstash-input-file*.

  Note:
  path indicates the file path, which must use UNIX separators, for example, *C:/test/multiline/*.log*. Otherwise, fuzzy match is not supported.

- **logstash-output-logservice**

  This plug-in is used to output the logs collected by the logstash-input-file plug-in to Log Service.

Parameters	Description
endpoint	Log Service endpoint. Example: *http://regionid.example.com*. For more information, see Log Service endpoint.
project	The project name of Log Service.
logstore	The Logstore name.
topic	The log topic name. The default value is null.

Parameters	Description
source	The log source. If this parameter is set to null, the IP address of the current machine is used as the log source. Otherwise, the log source is subject to the specified parameter value.
access_key_id	The AccessKey ID of the Alibaba Cloud account.
access_key_secret	The AccessKey Secret of the Alibaba Cloud account.
max_send_retry	The maximum number of retries performed when data packets cannot be sent to Log Service because of an exception. Data packets with retry failures are discarded. The retry interval is 200 ms.

Procedure

1. Create collection configurations

   Create a configuration file in the `C:\logstash-2.2.2-win\conf\` directory and then restart Logstash to apply the file.

   You can create a configuration file for each log type. The file name format is `*.conf`. For easier management, we recommend that you create all the configuration files in the `C:\logstash-2.2.2-win\conf\` directory.

   Note:
   The configuration file must be encoded in UTF-8 format without BOM. You can use Notepad++ to modify the file encoding format.

   - IIS logs

     For more information, see *Use Logstash to collect IIS logs*.

   - CSV logs

     Use the system time of log collection as the log uploaded time. For more information, see CSV log configuration.

   - Logs with built-in time

     Take CSV log format as an example. Use the time in the log content as the log uploaded time. For more information, see *Use Logstash to collect CSV logs*.

   - General logs

By default, the system time of log collection is used as the log uploaded time. Log fields are not parsed. Single-line logs and multiline logs are supported. For more information, see *Use Logstash to collect other logs*.

2. Verify configuration syntax

   1. Run `PowerShell` or `cmd.exe` to go to the Logstash installation directory:

      ```
      PS C:\logstash-2.2.2-win\bin> .\logstash.bat agent --configtest --
      config C:\logstash-2.2.2-win\conf\iis_log.conf
      ```

   2. Modify the collection configuration file. Temporarily add a line of rubydebug configuration in the output phase to output the collection results to the console. Set the type field as per your needs.

      ```
      output {
      If [type] = "***"{
        stdout { codec => rubydebug }
        logservice {

        }
      ```

   3. Run `PowerShell` or `cmd.exe` to go to the Logstash installation directory and start the process:

      ```
      PS C:\logstash-2.2.2-win\bin> .\logstash.bat agent -f C:\logstash-
      2.2.2-win\conf
      ```

   After the verification, end the *`logstash.bat`* process and delete the temporary configuration item rubydebug.

What's next

When *`logstash.bat`* is started in PowerShell, the Logstash process is working in the frontend. Logstash is generally used for testing configurations and debugging collections. Therefore, we recommend that you set Logstash as a Windows service after the debugging is passed so as to enable Logstash to work in the backend and start automatically when power-on. For how to set Logstash as a Windows service, see *Set Logstash as a Windows service*.

1.2.5 Advanced functions

Logstash provides *multiple plug-ins* to meet personalized requirements. For example:

- *grok*: Structurally parses logs into multiple fields by using regular expressions.
- *json_lines* and *json*: Structurally parses JSON logs.

- *date*: Parses and converts the date and time fields of logs.
- *multiline*: Customizes complex types of multiline logs.
- *kv*: Structurally parses logs of key-value pair type.

1.2.6 Logstash error processing

If you encounter the following collection errors when using Logstash to collect logs, follow the corresponding suggestions and process the errors.

If you encounter the following collection errors when using Logstash to collect logs, follow the corresponding suggestions and process the errors.

- Data with garbled characters in Log Service

  Logstash supports UTF-8 file encoding by default. Check whether input files are correctly encoded or not.

- Error message in the console

  The error `io/console not supported; tty will not be manipulated` is prompted in the console. However, the error does not affect the functions and can be ignored.

  If other errors occur, we recommend that you search Google or Logstash forums for help.

1.3 SDK collection

1.3.1 Producer Library

LogHub Producer Library is a LogHub class library written for high-concurrency Java applications. Producer Library and *Consumer Library* are the read and write packaging for LogHub to lower the threshold for data collection and consumption.

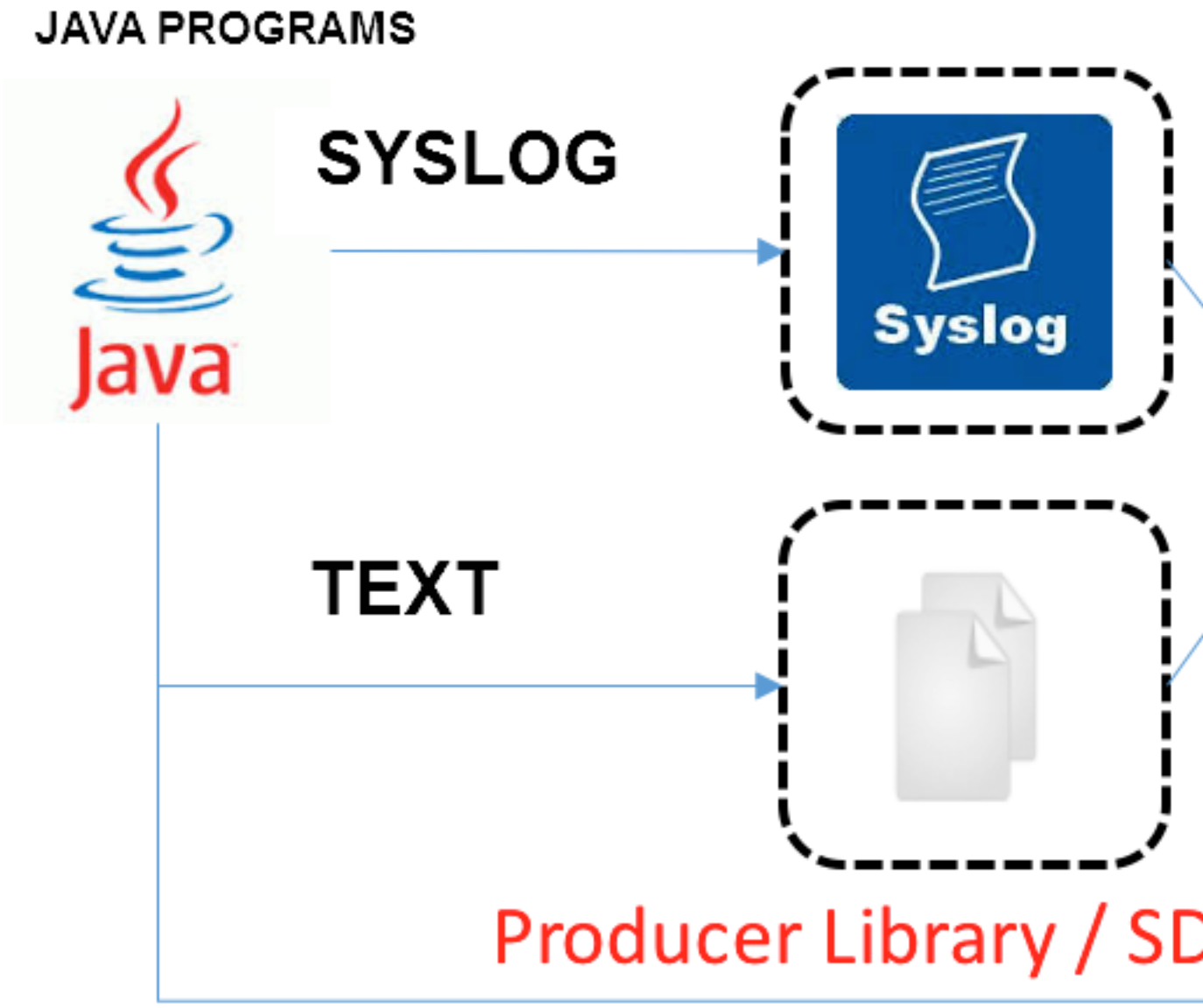
Function features

- Provides an asynchronous send interface to guarantee the thread security.
- Configurations of multiple projects can be added.
- The number of network I/O threads used for sending logs can be configured.
- The number and size of logs of a merged package can be configured.
- The memory usage is controllable. When the memory usage reaches your configured threshold value, the send interface of producer is blocked until idle memory is available.

Function advantages

- Logs collected from the client are not flushed into the disk. Data is directly sent to Log Service by using the network after being generated.
- High concurrency write operations on the client. For example, more than one hundred write operations are performed in one second.
- Client computing logically separated from I/O. Printing logs does not affect the computing time used.

In the preceding scenarios, Producer Library simplifies your program development steps, aggregates write requests in batches, and sends the requests to the LogHub server asynchrono usly. During the process, you can configure the parameters for aggregation in batches and the logic to process server exception.



Compare the preceding access methods:

Access method	Advantages/disadvantages	Scenario
Log flushed into the disk + Logtail	Log collection decoupled from logging, no need to modify the code.	Common scenarios

Access method	Advantages/disadvantages	Scenario
Syslog + Logtail	Good performance (80 MB/s). Logs are not flushed into the disk. The syslog protocol must be supported.	Syslog scenarios.
SDK direct transmission	Not flushed into the disk, and directly sent to the server. Switching between the network I/O and program I/O must be properly processed.	Logs are not flushed into the disk.
Producer Library	Not flushed into the disk, asynchronously merged and sent to the server, with good throughput.	Logs are not flushed into the disk and the client QPS is high.

Procedure

- *Java Producer*
- *Log4J1. Log4J1.XAppender (based on Java Producer)*
- *Log4J2. XAppender (based on Java Producer)*
- *LogBack Appender (based on Java Producer)*
- *C Producer*
- *C Producer Lite*

1.3.2 Log4j Appender

Log4j is an open-source project of Apache, which allows you to set the log output destination to console, file, GUI component, socket server, NT event recorder, or UNIX Syslog daemon. You can also set the output format and level of each log to control log generation with a finer granularity. These configurations can be performed flexibly by using a configuration file without modifying application codes.

Alibaba Cloud Log4j Appender allows you to set the log output destination to Alibaba Cloud Log Service. For more information about download link and user guide, refer to *Github*.

1.3.3 C Producer Library

Besides the Producer Library of Java version, LogHub also supports the Producer Library and Producer Lite Library of the C version, which provides you with a simple and high-performance one-stop log collection solution across platforms and with low consumption of resources.

For the GitHub project address, see:

- *C Producer Library (recommended for servers)*

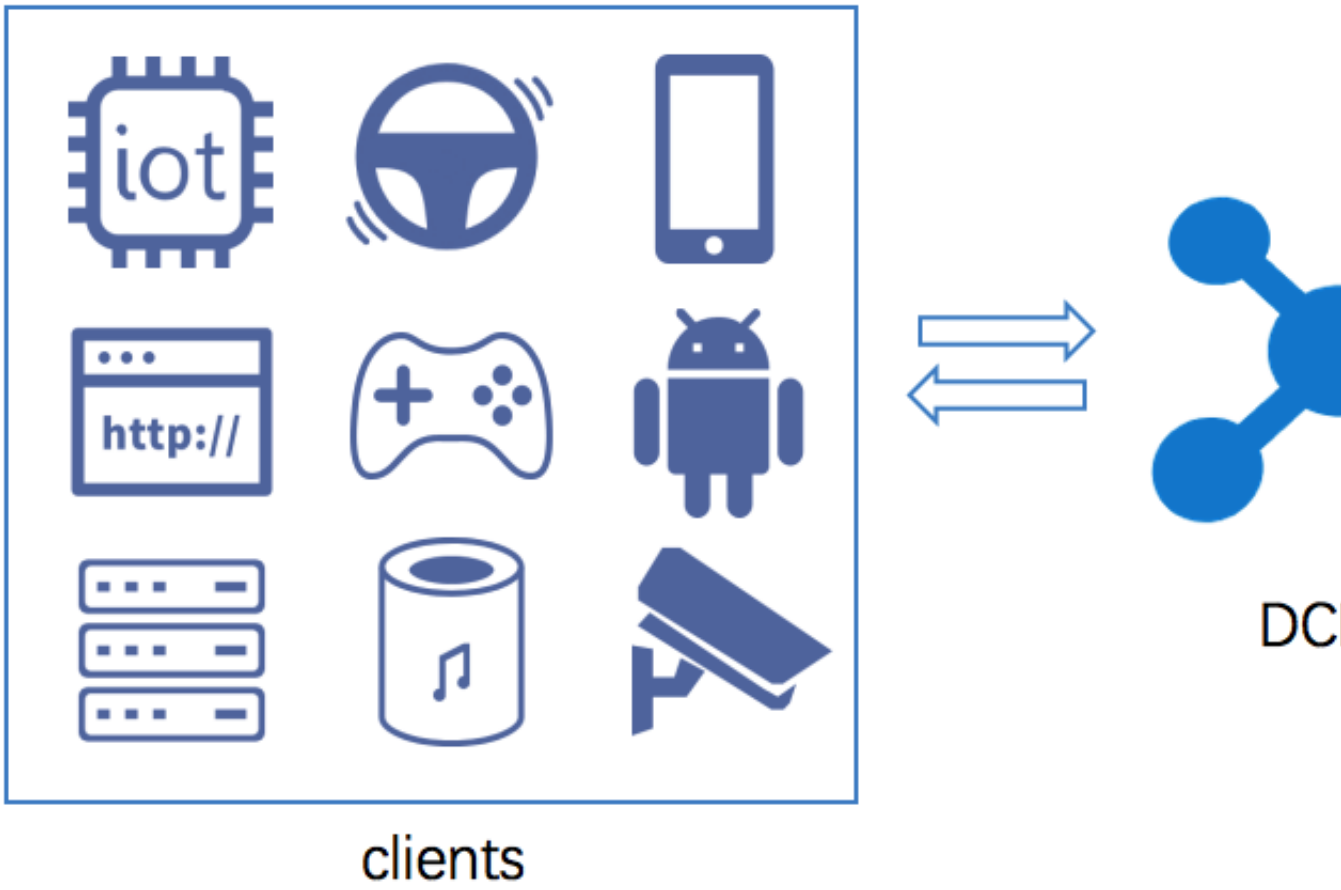
- *C Producer Lite Library (recommended for IOT and smart devices)*

1.4 Collection acceleration

1.4.1 Overview

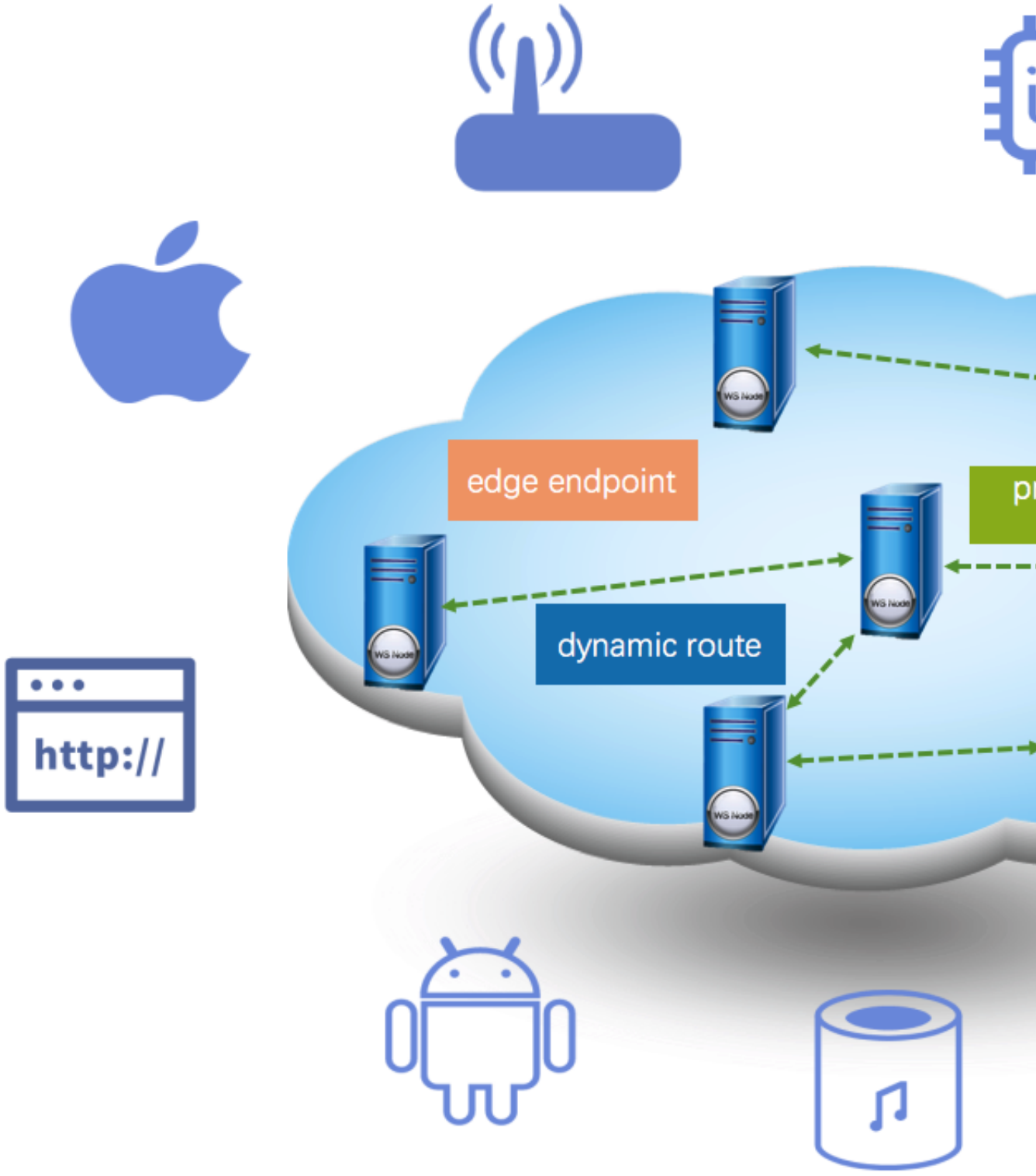
Log Service adds a network type of **Global Acceleration Public Network** on the basis of Virtual Private Cloud (VPC) and public network. Compared with the ordinary public network access, Global Acceleration Public Network has significant advantages in terms of delay and stability, and is suitable for scenarios with high demands for data collection, low consumption delay, and reliability. Global Acceleration for Log Service depends on the acceleration environment provided by Alibaba Cloud *Dynamic Route for CDN* products. This function improves overall site performance and user experience by solving problems of slow response, packet loss, and unstable services. These problems are caused by factors such as cross-carriers access, network instability, traffic spikes, and network congestion.

Global Acceleration for Log Service is based on Alibaba Cloud Content Delivery Network (CDN) hardware resources, and optimizes the stability of log collection and data transmission from various forms of data sources such as mobile phones, Internet of Things (IoT) devices, smart devices, self-built Internet Data Centers (IDCs), and other cloud servers.



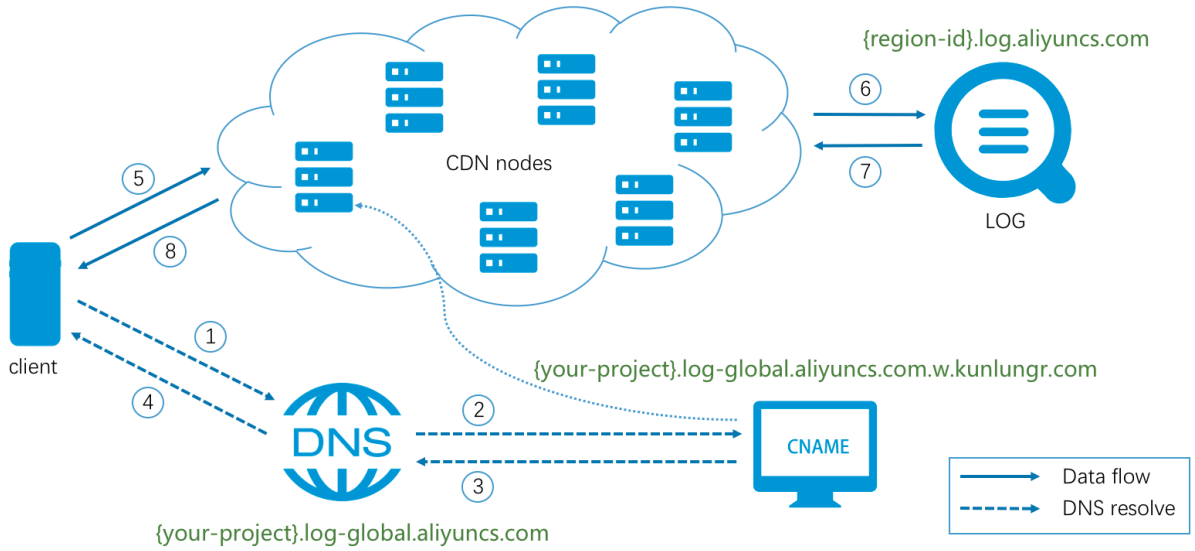
Technical principles

Global Acceleration for Log Service is based on Alibaba Cloud CDN hardware resources. Your global access terminals (such as mobile phones, IOT devices, smart devices, self-built IDCs, and other cloud servers), access the nearest edge node of Alibaba Cloud CDN all over the world and route to Log Service through CDN inner high-speed channels. Compared with common public network transmission, network delay and jitter can be reduced greatly in this method.



The processing flow of Global Acceleration requests for Log Service is shown in the preceding figure. The overall flow is detailed as follows:

1. The client needs to send a domain name resolution request to the public DNS before sending requests of log upload or log download to the Log Service acceleration domain name *your-project*.log-global.aliyuncs.com.
2. The domain name at the public DNS *your-project*.log-global.aliyuncs.com points to the CNAME address *your-project*.log-global.aliyuncs.com.w.kunlungr.com. The domain name resolution is forwarded to the CNAME nodes of Alibaba Cloud CDN.
3. Based on Alibaba Cloud CDN smart scheduling system, CNAME nodes return the IP address of the optimal CDN edge node to the public DNS.
4. The public DNS returns the IP address finally resolved to the client.
5. The client sends a request to the server based on the obtained IP address.
6. After receiving the request, the CDN edge node routes the request to the node closest to the Log Service server based on the dynamic route lookup and private transport protocol. Finally, the request is forwarded to Log Service.
7. After receiving the request from the CDN node, the server of Log Service returns the result of the request to the CDN node.
8. CDN transparently transmits the result or data returned by Log Service to the client.



Billing method

Global Acceleration costs for Log Service include:

- Costs for accessing Log Service

Costs for accessing Log Service is the same as that in common public network. Log Service supports **Pay-As-You-Go** billing method, and provides **FreeTier quota**. For more information, see *Billing method*.

- Service costs for Dynamic Route for CDN

  For information about cloud product costs of Dynamic Route for CDN, see *Billing Method of Dynamic Route for CDN*.

Scenarios

- Advertisement

  Log data about advertising browsing and clicking are extremely important for advertising billing . Advertising carriers include mobile terminal embedding, H5 pages, PC ends, and more all over the world. In some remote areas, the public network data transmission is less stable and risks of log loss exist. A more stable and reliable log upload channel can be obtained through Global Acceleration.

- Online game

  The online game industry has high requirements on the performance and stability of data collection in the official website, logon service, sales service, game service, and other services . The timeliness and stability of data collection are hard to be guaranteed in the case of mobile game data collection and data back transmission from globalized games. We recommend that you use Global Acceleration for Log Service to solve the preceding issues.

- Finance

  Financial-related applications require high availability and high security for network. Audit logs of each transaction and each user action must be collected securely and reliably to the server. At present, mobile transactions have become mainstream. For example, online banking, credit card malls, mobile securities, and other types of transactions can achieve secure, fast, and stable log collection by using HTTPS Global Acceleration for Log Service.

- Internet of Things

  IoT devices and smart devices (for example, smart speakers and smart watches) collect sensor data, operation logs, critical system logs, and other data to the server for data analysis. These devices are usually distributed all over the world and the surrounding network is not always reliably. To achieve stable and reliable log collection, we recommend using Global Acceleration for Log Service.

Acceleration effect

Region	Delay ms (common public network)	Delay ms (acceleration)	Time-out ratio % (common public network)	Time-out ratio % (acceleration)
Hangzhou	152.881	128.501	0.0	0.0
Europe	1750.738	614.227	0.5908	0.0
USA	736.614	458.340	0.0010	0.0
Singapore	567.287	277.897	0.0024	0.0
Middle East	2849.070	444.523	1.0168	0.0
Australia	1491.864	538.403	0.014	0.0

The test environment is as follows:

- Region of Log service: North China 5 (Hohhot)
- Average upload packet size: 10KB
- Test time range: one day (average)
- Request type: HTTPS
- Request server: Alibaba Cloud ECS (Specification 1C1GB)

Note:
The acceleration effect is for reference only.

1.4.2 Enable Global Acceleration

To enable Global Acceleration for Log Service, see the following steps.

Prerequisite

- You have enabled Log Service and created the project and Logstore.
- You have enabled *Dynamic Route for CDN*.
- To *Enable HTTPS acceleration*, *Enable HTTP acceleration* first.

Configuration

After HTTP Global Acceleration is enabled for the project, you can also configure Global Acceleration of Logtail, SDK, and other methods according to your needs.

1. *Enable HTTP acceleration*.
2. Enable Global Acceleration of Logtail, SDK, and other methods.

- HTTPS

  If you use HTTPS to access Log Service, make sure that HTTPS acceleration is enabled. To configure HTTPS acceleration, see *Enable HTTPS acceleration*.

- Logtail log collection

  When you install Logtail, select the **Global Acceleration** network type at the page prompt. Then you can obtain global acceleration when you collect logs by using Logtail.

- SDK, Producer, and Consumer

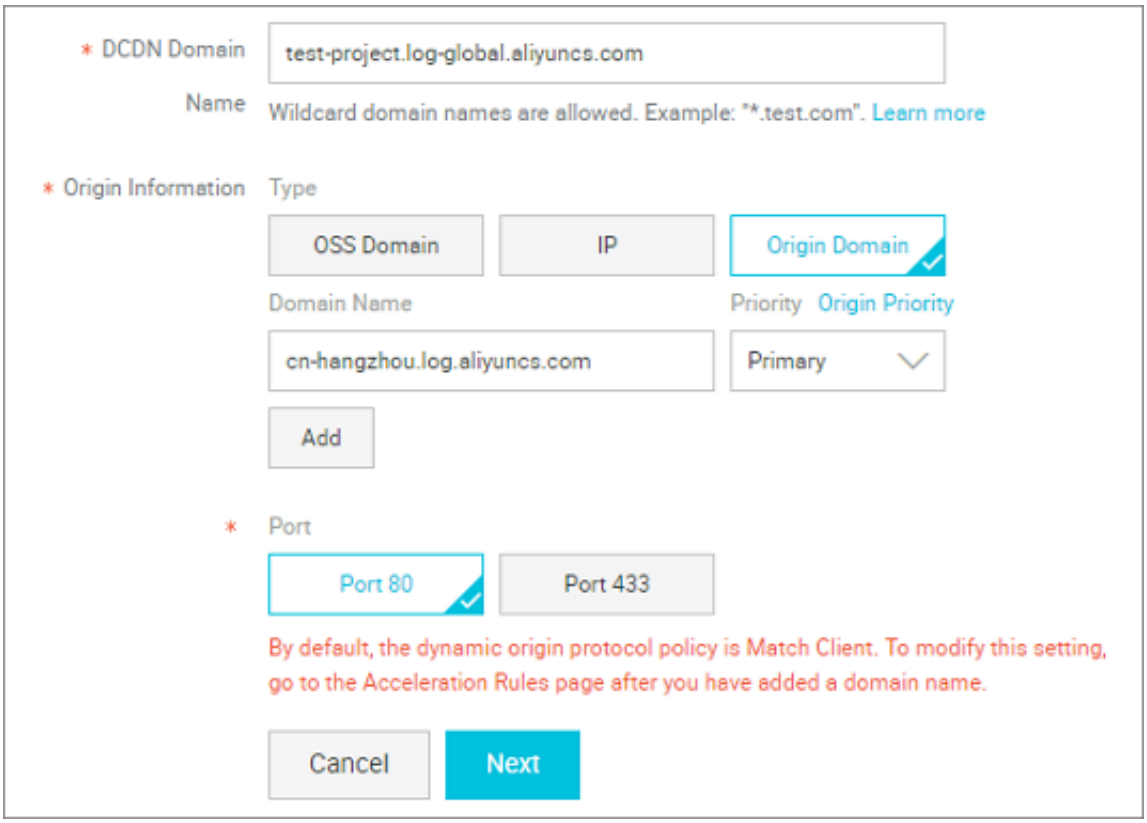
  Other ways to access Log Service such as SDK, Producer, and Consumer, can be accelerated by replacing the endpoint with `log-global.aliyuncs.com`.

Enable HTTP acceleration

1. Log on to the *Dynamic Route for CDN Console*. Click **Domain Names** in the left-side navigation pane to enter the **Domain Names** page.
2. Click **Add Domain Name** in the upper left corner to enter the **Add Domain Name** page.
3. Enter the **`DCDN Domain`** and other information, and click **Next**.

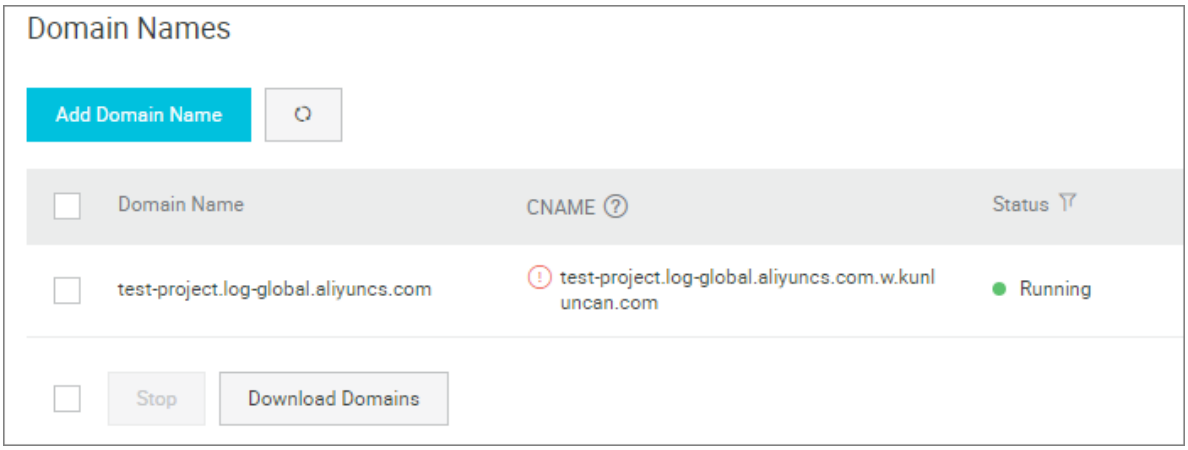
Configuration	Description
Accelerated domain name	*`project_name`*`.log-global.aliyuncs.com` Replace *`project_name`* with your project name.
Origin site type	Select `Origin Domain`.
Domain name	Enter the public network endpoint for the region to which your project belongs. For information about endpoints, see *Service endpoint*.
Port	Please select port 80. If you have an HTTPS acceleration requirement, see **Enable HTTPS acceleration**.
Accelerated area	By default, this configuration item is not displayed and the acceleration area is `Domesticate acceleration`. If you have a demand for Global Acceleration, open a ticket for the Dynamic Route for CDN product to apply for a whitelist . After your application is approved, you can select an acceleration region based on your needs.

For more information about adding domain names, see 8.

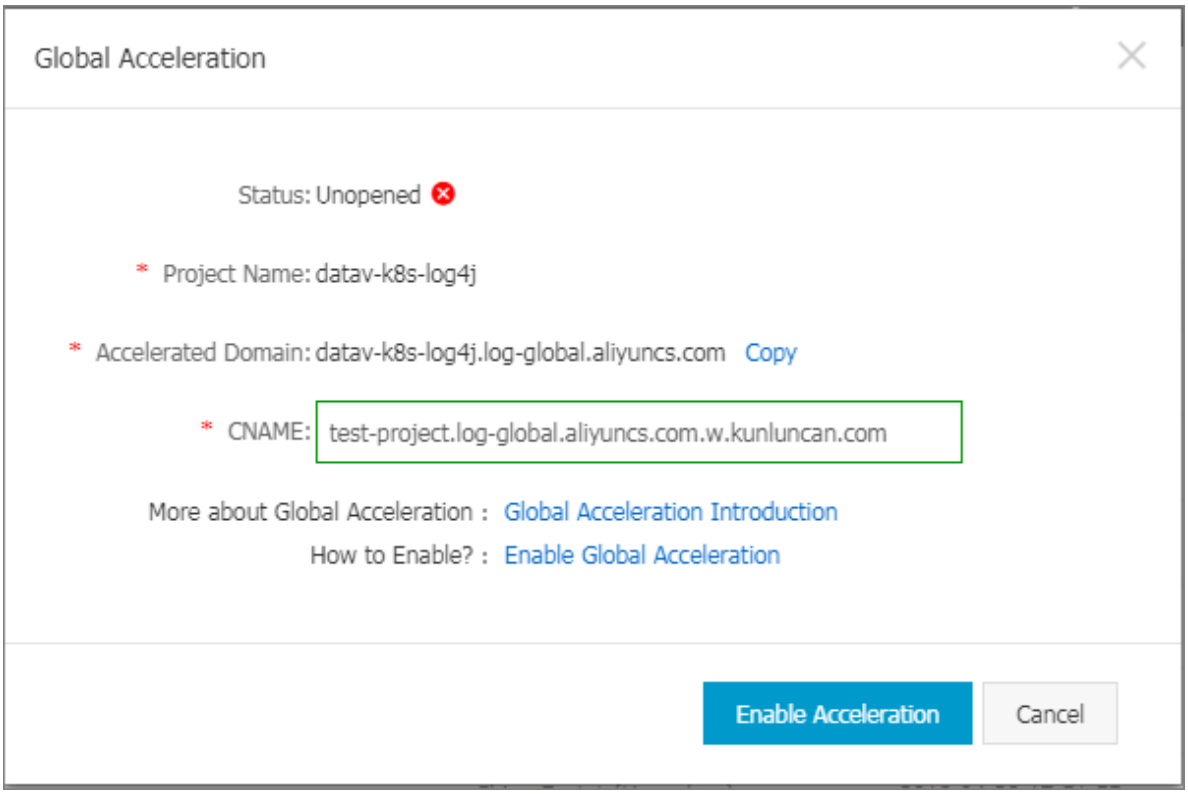


4. Go to the **Domain management** page as prompted.

   You can view the **CNAME** of each corresponding domain name in the **Domain name management** page.



5. Log on to the Log Service console and click **Global Acceleration** at the right of a specified project in the **Project list**.
6. Enter the **CNAME** corresponding to the accelerated domain name in the dialog box. Click **Enable acceleration**.

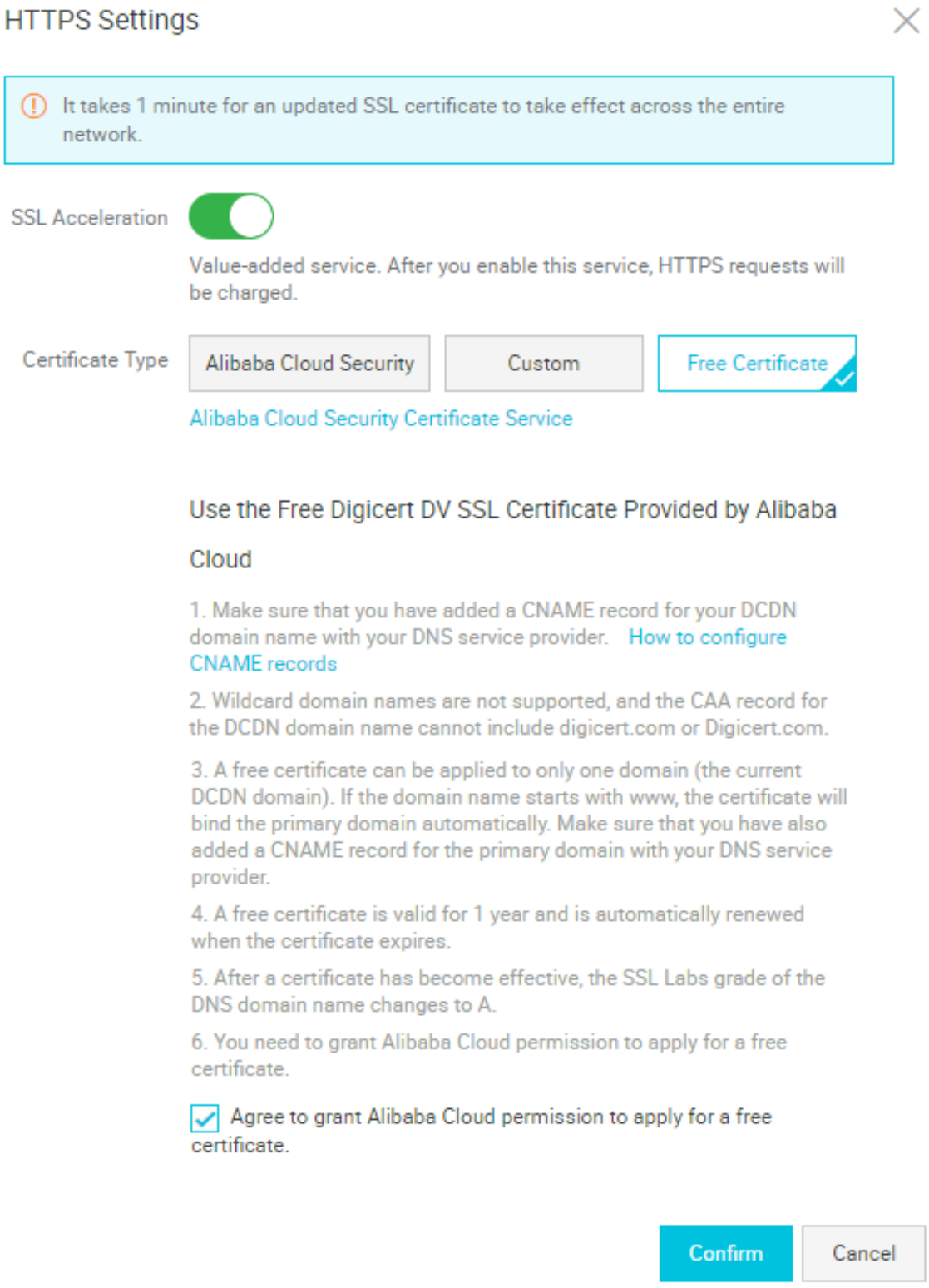


After you complete the preceding steps, Global Acceleration for Log Service is enabled.

Enable HTTPS acceleration

After enabling HTTP acceleration, if you have HTTPS access requirements, you can use the following steps to enable HTTPS acceleration.

1. Log on to the *Dynamic Route for CDN Console*. Click **Domain Names** in the left-side navigation pane to enter the **Domain Names** page.
2. Click **Configure** to the right of a specified domain name.
3. Click **HTTPS Settings** in the left-side navigation pane and click **Modify** in the column of **SSL Certificate** to enter the **HTTPS Settings** page.
4. Configure **`SSL Acceleration`** and **`Certificate Type`**.
   - Enable **`SSL Acceleration`**.
   - Select `Free Certificate` for **`Certificate Type`**.



After the configuration is completed, select **Agree to grant Alibaba Cloud permission to apply for a free certificate.**, and click **Confirm**.

For more information about HTTPS settings, see *HTTPS##*.

Verify if the acceleration configuration takes effect

FAQ

- **How to verify if the acceleration configuration takes effect** ?

  After the configuration is completed, you can verify if the acceleration takes effect by accessing your accelerated domain name.

  For example, if Global Acceleration is enabled for the *test-project* project, you can use `curl` to send a request to the accelerated domain name. If the following type of output is returned, the acceleration takes effect.

  ```
  $curl test-project.log-global.aliyuncs.com
  {"Error":{"Code":"OLSInvalidMethod","Message":"The script name is
  invalid : /","RequestId":"5B55386A2CE41D1F4FBCF7E7"}}
  ```

  For more information about checking methods, see *How to verify if the acceleration takes effect* .

- **How to handle the error of** `project not exist` reported in accessing accelerated domain name?

  This problem is caused usually by an invalid source site address. Log on to the Dynamic Route for CDN console and change the source site address to the public network address of the region to which your project belongs. For information about address list, see *Service endpoint*.

  Note:
  Changing the source site address has a synchronization delay of several minutes.

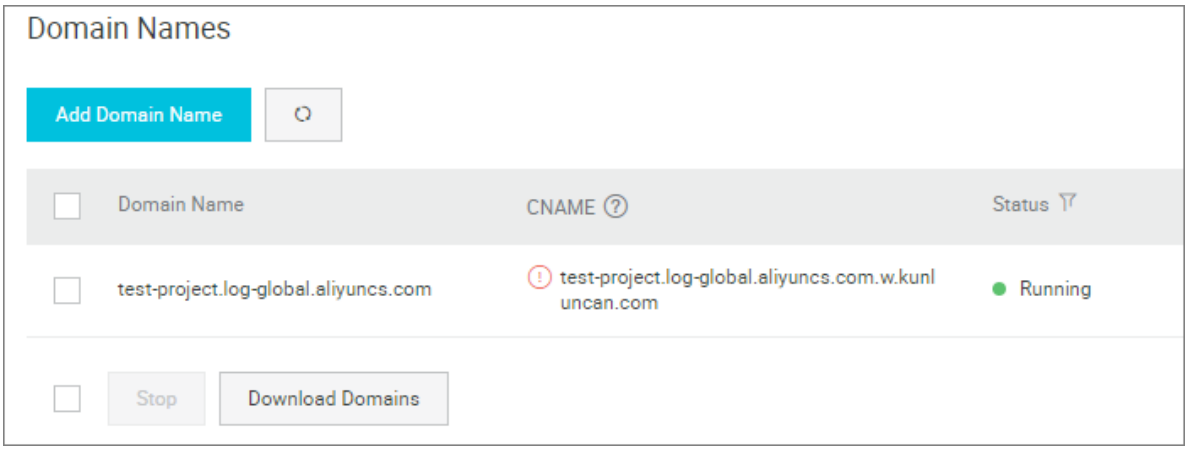
1.4.3 Disable Global Acceleration

To disable Global Acceleration for Log Service, perform the following operations.

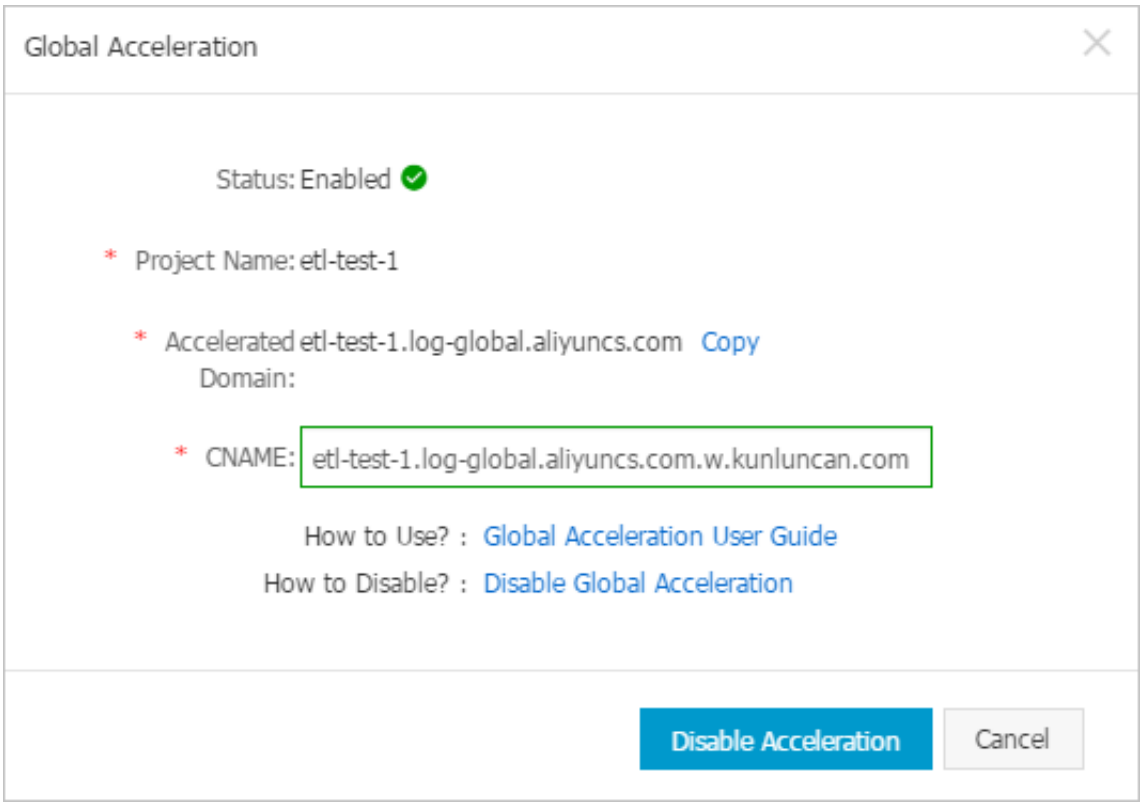
Note:
When you disable Global Acceleration, the accelerated domain name configured during provisioning becomes unavailable. Make sure that all of your clients do not upload or request data through the domain name before you disable Global Acceleration.

Disable Global Acceleration

1. Log on to the *Dynamic Route for CDN Console*. Click **Domain name management** in the left-side navigation pane to enter the **Domain name management** page.
2. View the **CNAME** corresponding to the domain name that is to be disabled .



3. Log on to the Log Service console. On the **Project list** page, click **Global Acceleration** at the right of a specified project.
4. Enter **CNAME** and click **Disable acceleration**.



2 Logtail collection

2.1 Limits

Table 2-1: Limits on file collection

Item	Capabilities and limits
File encoding	Log files encoded in UTF-8 and GBK are supported. Log files encoded in other formats result in undefined behaviors such as gibberish and data loss. We recommend that you use UTF-8 encoding for better processing performance.
Log file size	Unlimited.
Log file rotation	Both `.log*` and `.log` are supported.
Log collection behavior upon log parsing block	When block occurs in log parsing, Logtail keeps the open status of the log file FD. If log file rotation occurs multiple times during the block, Logtail attempts to keep the log parsing sequence of each rotation. If the number of unparsed log rotations is more than 20, Logtail does not process subsequent log files. Soft link support More information, see here.
Single log size	Monitored directories can be soft links.
Single log size	The size of a single log cannot exceed 512 KB . If multiple-line logs are divided by a regular expression, the maximum size of each log is still 512 KB. If the log size exceeds 512 KB , the log is forced to be divided into multiple parts for collection. For example, a log is 1025 KB. The first 512 KB is processed for the first time, the subsequent 512 KB is processed for the second time, and the last 1 KB is processed for the third time.
Regular expression type	Use regular expressions that are compatible with Perl.
Multiple collection configurations for the same file	Not supported. We recommend that you collect log files to a Logstore and configure multiple subscriptions. If this function is required,

Item	Capabilities and limits
	configure a soft link for the log file to bypass this limit.
File opening behavior	Logtail keeps a file to be collected in the open status. Logtail closes the file if the file does not have any modification within five minutes.
First log collection behavior	Logtail only collects incremental log files. If modifications are found in a file for the first time and the file size exceeds 1 MB, Logtail collects the logs from the last 1 MB. Otherwise, Logtail collects logs from the beginning. If a log file is not modified after the configuration is issued, Logtail does not collect this file.
Non-standard text log	For a row containing '\0' in the log. The log is truncated to the first '\0'.

Table 2-2: Checkpoint management

Item	Capabilities and limits
Checkpoint timeout period	If the file has not been modified for more than 30 days, the Checkpoint is deleted.
Checkpoint storage policy	Regular save every 15 minutes, automatically saved when the program exits.
Checkpoint save path	The default save path is `/tmp/logtail_checkpoint` , you can modify the parameters according to *Configure startup parameters*.

Table 2-3: Limits on configuration

Item	Capabilities and limits
Configuration update	Your updated configuration takes effect with a delay of about 30 seconds.
Dynamic configuration loading	Supported. The configuration update does not affect other collections.
Number of configurations	Theoretically unlimited. We recommend that the number of collection configurations for a server is no more than 100.
Multi-tenant isolation	The isolation between collection configurations.

Table 2-4: Limits on resources and performance

Item	Capabilities and limits
Log processing throughput	The default limit to raw log traffic is 2 MB/s. Data is uploaded after being encoded and compressed, generally with a compression ratio of 5–10 times. Logs may be lost if the log traffic exceeds the limit. To adjust the parameter, see *Configure startup parameters* Configure startup parameters.
Maximum performance	In case of single core, the maximum processing capability is 100 MB /s for logs in simple mode, 20 MB/s by default for logs in full mode (depending on the complexity of the regular expression), 40 MB/s for logs in delimiter mode, and 30 MB/s for logs in JSON mode. Enabling multiple log processing threads improves the performance by 1.5–3 times.
Number of monitored directories	Logtail actively limits the depth of monitored directories to conserve your resources. If the upper limit is reached, Logtail stops monitoring more directories and log files. Logtail monitors at most 3,000 directories (including subdirectories).
Default resource limit	By default, Logtail occupies up to 40% of CPU usage and 256 MB of memory usage. If logs are generated at a high speed, you can adjust the parameter by following the *Configure startup parameters* Configure startup parameters.
Processing policy for resource limit exceeding	If the resources occupied by Logtail in 3 minutes exceed the upper limit, Logtail is forced to restart, which may cause loss or duplication of data.

Table 2-5: Limits on error handling

Item	Capabilities and limits
Network error handling	If the network connection is abnormal, Logtail actively retries and automatically adjusts the retry interval.
Handling of resource quota exceeding	If the data transmission rate exceeds the maximum quota of Logstore, Logtail blocks log collection and automatically retries.
Maximum retry period for timeout	If data transmission fails for more than 6 successive hours, Logtail discards the data.
Status self-check	Logtail automatically restarts in the case of an exception, for example, abnormal exit of a program or resource limit exceeding.

Table 2-6: Other limits

Item	Capabilities and limits
Log collection delay	Except for block status, the delay in log collection by Logtail does not exceed one second after logs are flushed to a disk.
Log uploading policy	Logtail automatically aggregates logs in the same file before uploading them. Log uploading is triggered in the condition that more than 2,000 logs are generated, the log file exceeds 2 MB, or the log collection exceeds 3 seconds.