

Log Service

FAQ

FAQ

What is Log Service?

Log Service is a platform service used to collect, store, and subscribe to logs. The service enables you to collect in real time, centrally manage and consume various types of logs.

What functions does Log Service provide?

- Multiple ways of logging (using APIs, SDKs, and Logtail)
- Definition of log collection and parsing methods using Logtail according to your need
- Management of log collection on thousands of machines in machine groups
- Real-time log consumption and subscription
- Simple configuration through the console, allowing you to perform all operations through the Web
- Seamless interconnection between the background and Alibaba Cloud products

How do I activate Log Service?

Currently, Log Service is in the open public beta stage. Visit www.aliyun.com, log on to your Alibaba Cloud account, go to the Log Service product page, and click "Activate Now" .

What are the basic concepts of Log Service?

- Core concepts: project (basic unit of log management), LogStore, shard, topic (applicable to level-2 LogStore classification), log (number of logs), and LogGroup
- Concepts about log collection: Logtail configuration (which defines how to collect logs) and machine group (used to manage machines in groups)

What are the components of Log Service?

Log Service consists of a log collection agent, a server, and other systems. Currently, the log collection agent is Logtail compatible with Windows and Linux. The server is responsible for reading, writing, and configuring Log Service APIs. Other systems include MaxCompute, to which Log Service synchronizes logs. Log Service will include the ability to synchronize logs to OSS and AMR for consumption.

How is a log defined in Log Service?

A log contains three parts: time (mandatory), log content (which consists of key– value pairs), and metadata (data source, which can be defined by the IP addresses of logs).

How does Log Service work with other Alibaba Cloud services?

- ECS: You can install Logtail (agent) to collect logs on your ECS server.
- MaxCompute: Logs collected to Log Service can be automatically posted to your MaxCompute table for online query and offline analysis.

Log management

How does the Log Service store and manage user logs?

LogStores are the basic units for storing and querying logs in the Log Service. Each LogStore stores a specific type of logs. Currently, you can add, delete, modify, and query LogStores on the Log Service Console or using APIs. After creating a LogStore, you can write logs into it through APIs or SDKs. The Log Service provides Logtail to collect logs on Alibaba Cloud ECS servers in an easy way.

Are logs lost after I delete a LogStore?

Yes. Exercise caution for LogStore deletion.

What is the log storage period of the Log Service? Can I modify this period?

The following three functions of the Log Service are related to the log storage period:

- LogHub: provides temporary storage for recent 48 hours to support real-time log consumption. Currently, the storage period cannot be modified.
- LogShipper: After logs are posted to OSS and MaxCompute, you can set a lifecycle in these products.
- LogSearch: supports storage periods of seven days, 30 days, and 90 days. The storage period is unchangeable once created. Because log indexes are hot data, select OSS or MaxCompute for a longer period of cold storage.

Can I share logs with other users?

Currently, the Log Service supports RAM, allowing you to share data through subaccount

authorization.

Log Collection

What types of logs does the Log Service collect?

The Log Service supports timestamped text logs encoded in UTF-8 that are generated within the past seven days and are no more than 15 minutes later than the current time.

In what ways does the Log Service collect logs? How do I choose among them?

The Log Service supports direct data writing using APIs (SDKs are currently available in four languages: Java, Python, PHP, and C#). It provides Linux- and Windows-compatible Logtail used to collect real-time updated logs from disk files.

1. If the logs generated by application programs are not flushed into disk, those logs can be written directly to the Log Service through APIs.
2. Logs that are written into disks in real time can be collected by Logtail.

How does the Log Service collect logs from ECS?

You can use Logtail to collect the ECS logs that are flushed into disk as follows:

1. Create a Logstore on the Log Service Console.
2. Perform Logtail configuration.
3. Create a machine group.
4. Execute the installation script to install the Logtail agent.
5. Apply the Logtail configurations to the machine group.

Does the Log Service collect historical logs?

You can only write data generated during the past seven days using an API. However, Logtail does not support historical data collection for the moment.

What data collection capability does the Log Service provide? Does it have any limitation?

You can adjust the number of shards in a Logstore as needed. Logtail collects data at a maximum rate

of 1 MB/s on ECS.

What should I pay attention to when using Logtail to collect logs on NAS?

For collection of Nginx access logs, the Nginx configurations of web servers are the same. Logs are written into files with the same name on different machines. (In this case, Logtail collects logs properly.) When NAS is used, Logtail may have missing logs or encounter a collection error if the Nginx logs on multiple machines are written into the same file on NAS (concurrent write to the same file). To avoid this problem, ensure that the logs on different web servers are written into different files on NAS.

What is Logtail?

Logtail is a log collection agent provided by the Log Service. Once installed on your machine, Logtail monitors specified log files and automatically uploads the new logs written into these files to your designated LogStore.

Does Logtail collect static log files?

Logtail monitors file changes based on change events in the file system and sends logs generated in real time to the Log Service. Logtail does not collect the content of unchanged logs.

What platforms does Logtail support?

Currently, Logtail supports 64-bit Linux and 32/64-bit Windows Server 2003-2012 system.

Linux:

- Alibaba Cloud Linux
- Ubuntu
- Debian
- CentOS
- OpenSUSE

Windows:

- Windows 7 (Client) 32bit
- Windows 7 (Client) 64bit
- Windows Server 2003 32bit
- Windows Server 2003 64bit
- Windows Server 2008 32bit
- Windows Server 2008 64bit

- Windows Server 2012 64bit

How do I install and upgrade the Logtail agent?

Installation: Install the Logtail agent using an installation script. Upgrade: The Log Service regularly upgrades the Logtail agent without interrupting the data collection process.

How do I configure the Logtail agent?

Refer to Logtail collection configuration on the console.

How does Logtail work?

1. On the console, configure the directory you want to monitor, the name of the log file, and the related parsing rule (regular expression).
2. When a log file is changed on your machine, Logtail receives an event from the file system and reads the new log.
3. Logtail parses the log format based on the regular expression and sends the log to the Log Service.

Does Logtail support log rotation?

When the log file a.LOG reaches a given size or lasts for a given period of time since created, a.LOG is renamed a.LOG.1 (or another name). A new a.LOG file is created for writing new logs. This process is called rotation. Logtail automatically rotates logs based on event notifications from the file system.

How does Logtail handle network exceptions?

In the case of a network exception or write quota overrun, Logtail caches collected logs to the local disk and resends those logs later. The maximum disk cache capacity is 500 MB. Newly cached data overwrites the old one when the 500-MB limit is exceeded. Cached files that fail to be sent to the Log Service within 24 hours are automatically deleted.

What is the log collection delay of Logtail?

Logtail collects logs based on events and sends collected logs to the Log Service within 3s.

How does Logtail process historical logs?

Logtail only collects real-time logs. If the logging time is more than 5 minutes different from the system time at which Logtail processes the log, the log is regarded as a historical log.

How long does a change in log collection configuration take effect for the Log Service?

After you apply configurations to a machine group on the console, Logtail loads and applies the configurations in 3 minutes or less.

How do I locate any log collection problems of Logtail?

1. Check whether the Logtail heartbeat is normal. If it is abnormal, **reinstall Logtail**.
2. Check whether the log files in log collection configuration are generated in real time.
3. Check whether the regular expression in log collection configuration matches the log content. If the regular expression does not match, view the error in the Logtail run log (Linux:/usr/local/ilogtail/ilogtail.LOG).

Why is the Logtail heartbeat abnormal?

1. Currently, the Logtail agent only supports 64-bit Linux operating systems.
2. Use **LogStash** to collect logs in a Windows system.

If the Logtail heartbeat is abnormal, follow these steps below to perform diagnosis.

- Check whether the Logtail process exists by running the following command. If the process does not exist, **reinstall Logtail**. If it exists, go to the next step.

```
sudo /etc/init.d/ilogtaild status
```

Run the following commands to check network connectivity:

Classic network

```
telnet logtail.cn-<region>-intranet.log.aliyuncs.com 80
```

VPC

```
telnet logtail.cn-<region>-vpc.log.aliyuncs.com 80
```

If your machine is not connected, perform the following check:

1. If the machine is configured with host name binding (run the hostname command to view the host name; the related file is /etc/hosts), check whether the bound IP address is the

same as that in the Log Service machine group.

2. If no host name is bound, check whether the IP address of the machine's first network adapter is the same as that in the Log Service machine group.

If the machine is not connected, the Log Service cannot receive heartbeat packets from the machine. In this case, contact the Log Service technical support team for troubleshooting.

If the problem persists, submit a ticket in the ticket system. The Log Service technical support team will look into the problem.

If Logtail machine group heartbeat is abnormal when collect logs using Logtail, you can identify a problem using Logtail checking tool or manual diagnosis.

Automatic diagnosis

Log Service supports Logtail checking tool.

If the checking result is normal, check step 3–6 of **Manual diagnose** based on the echo message of checking result.

Manual diagnose

Logtail machine group heartbeat failed is usually caused by the following reasons, please inspect one by one.

1. Network disconnected

Execute the following command to check the network connectivity, and make sure the network is running normally.

Classic network

```
telnet logtail.cn-<region>-intranet.log.aliyuncs.com 80
```

VPC

```
telnet logtail.cn-<region>-vpc.log.aliyuncs.com 80
```

Internet

```
telnet logtail.cn-<region>.log.aliyuncs.com 80
```

2. Logtail process does not exists

Check whether the Logtail process exists by running the following command. If it does not exist, install Logtail. Identify the region where your project is located and check whether the region

matches that in config. If Logtail process exists, go to the next step.

Linux:

```
sudo /etc/init.d/ilogtaild status
```

Windows:

Control Panel -> Management Tool -> Service
Check LogtailDaemon、LogtailWorker running status.

3. Parameters error

You need to specify the right endpoint for your Logtail. Check the parameters you have already set:

- Linux: /usr/local/ilogtail/ilogtail_config.json
- Windows x64: C:\Program Files (x86)\Alibaba\Logtail\ilogtail_config.json
- Windows x32: C:\Program Files\Alibaba\Logtail\ilogtail_config.json

Make sure that:

- The project is located in the same region with your Logtail Endpoint.
- You have already chosen the same domain name based on network environment of your servers. An internal domain name selected in VPC environment may fail to be connected. Telnet to the domain name that you configured in ilogtail_config.json, such as telnet logtail.cn-hangzhou-intranet.log.aliyuncs.com 80.

4. Logtail was configured Wrong IP or user ID

Usually:

- The IP address is only a tag and does not affect network access. The method of setting this IP address is as follows: Obtain the bound IP address in /etc/hosts. Run the hostname command to set machinename.
- If no IP address is bound, obtain the IP address of the first network adapter (ifconfig eth0).

Check IP address on servers:

- Linux: /usr/local/ilogtail/app_info.json
- Windows x64: C:\Program Files (x86)\Alibaba\Logtail\app_info.json
- Windows x32: C:\Program Files\Alibaba\Logtail\app_info.json

If the IP address filled in machine group console is different with that of Logtail obtained, modify it depend on these circumstances:

- If the IP address filled in machine group console is wrong, please correct it and save the

configuration.

- If you changed the network configuration (such as /etc/hosts), restart Logtail to obtain the new IP address.

If necessary, you can execute the following command to restart Logtail.

- Linux: `sudo /etc/init.d/ilogtaild stop; sudo /etc/init.d/ilogtaild start`
- Windows: **Control Panel -> Management Tool -> Service -> Restart LogtailWorker**

5. AccessKey is not configured

Check /usr/local/ilogtail/ilogtail.LOG if there is a mistake:Unauthorized ErrorMessage:no authority, denied by ACL

If the above error occurs, your main account is not configured AccessKey, so Logtail does not work. Refer to 5 Minute Quick Start to configure AccessKey steps to properly configure AccessKey.

6. The server is non-Alibaba Cloud ECS or not belong to the same account with the current Project of Log Service

As shown below, these are two types of situations that servers installed Logtail must be authenticated to collect logs. For more information, refer to aliquid.

1. The server is non-Alibaba Cloud ECS
2. The server is not belong to the same account with the current Project of Log Service

If the problem persists, submit a ticket in the ticket system. The Log Service technical support team will look into the problem. Please provide your Project name, Logstore name, machine group, app_info.json, ilogtail_config.json and the checking results of Logtail checking tool.

Log Service LogHub and Kafka

Kafka is a distributed messaging system with high throughput and horizontal scaling and is widely used for message publishing and subscription. It is available as open source software. You can build a Kafka cluster as needed.

The Log Service is a log-specific platform service built upon Apsara Pangu. It supports the real-time collection, storage, distribution, and query of all types of logs. The Log Service uses standard RESTful APIs.

The Log Service LogHub provides public channels of log collection and distribution, removing the need to build and maintain your Kafka cluster.

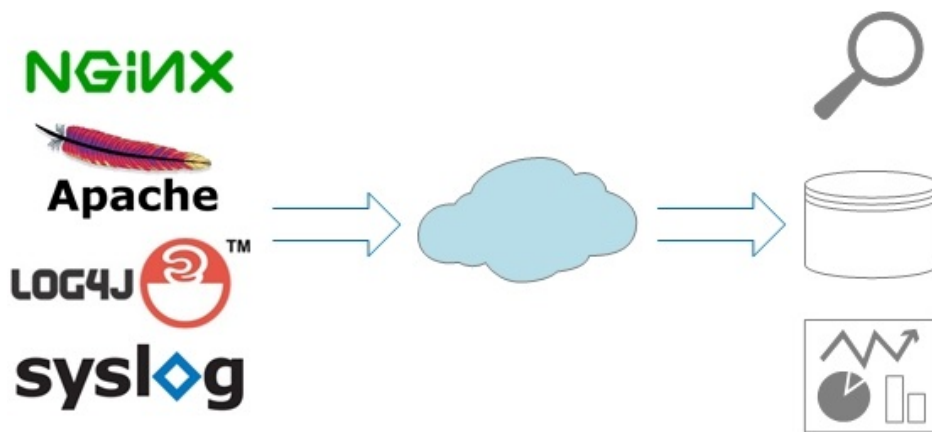
Mapping between Log Service LogHub and Kafka

| Concept | Kafka | LogHub |
|---------------------------|-----------|----------|
| Storage object | Topic | LogStore |
| Horizontal partitioning | Partition | Shard |
| Data consumption location | Offset | Cursor |

Functional comparison between LogHub and Kafka

| Function | Kafka | LogHub |
|------------------------------|--|---|
| Use of dependency | Self-built or shared Kafka cluster | Log Service |
| Communications protocol | Network interconnection over TCP | HTTP (RESTful API), Port 80 |
| Access control | Unavailable | Signature authentication+Access control based on an Alibaba Cloud account |
| Dynamic resizing | Unavailable for the moment | Auto scaling (merge/split) of shard quantities in a dynamic manner without impact |
| Multi-tenant QoS | Unavailable for the moment | Shard-based standard throttling |
| Data copy quantity | Custom | Unavailable for the moment; three copies by default |
| Failover/Replication | Completed by tools | Completed in an automatic and perception-free manner |
| Resizing/Upgrade | Completed by tools with service impact | Perception-free |
| Write mode | Round robin/Key hash | Currently, only round robin and key hash are supported. |
| Current consumption location | Stored in the Zookeeper of the Kafka cluster | Maintained at the service end without user intervention |

Billions of servers, mobile terminals, and network devices generate massive logs every day in the DT era. Centralized log processing effectively supports log consumption during the entire lifecycle. The first step of log processing is to store logs collected from devices in the cloud.



Three log collectors

logstash

- LogStash is represented by the letter “L” in the ELK stack in the open source community. It plays an active role and supports many plug-ins in the ecosystem.
- LogStash is implemented based on JRuby and runs across different platforms on JVM.
- Its modular design delivers high scalability and operability.

fluentd

- Fluentd is a popular log collector in the open source community. It is commercially available as td-agent and maintained by Treasure Data. td-agent is evaluated in this document.
- Fluentd is implemented based on CRuby and delivers good performance by re-implementing the components essential for performance using the C language.
- Fluentd features concise design and provides reliable pipelines for data transfer.
- Compared to LogStash, Fluentd supports fewer plug-ins.

- logtail

- Logtail is the producer in the Alibaba Cloud Log Service. It has been widely applied in the big data field by Alibaba for more than three years.
- Logtail is implemented using the C++ language and delivers high performance after great efforts have been made to improve its stability, resource control capability, and management.
- Compared to the community support of LogStash and Fluentd, Logtail is dedicated to log collection with lower functional variety.

Function comparison

| Function | LogStash | Fluentd | Logtail |
|-------------|----------|---------|-----------------|
| Log reading | Polling | Polling | Event triggered |

| | | | |
|--------------------------------------|---|---|---|
| File rotation | Supported | Supported | Supported |
| Failover (local checkpoint) | Supported | Supported | Supported |
| General log parsing | Grok parsing (based on a regular expression) | Parsing based on a regular expression | Parsing based on a regular expression |
| Specific log type | Support of delimiter, key-value, JSON, and other mainstream formats | Support of delimiter, key-value, JSON, and other mainstream formats | Support of delimiter, key-value, JSON, and other mainstream formats |
| Data compression before sending | Supported by a plug-in | Supported by a plug-in | LZ4 |
| Data filter | Supported | Supported | Supported |
| Buffer-based data transfer | Supported by a plug-in | Supported by a plug-in | Supported |
| Transfer exception handling | Supported by a plug-in | Supported by a plug-in | Supported |
| Runtime environment | JRuby implementation with JVM environment dependency | CRuby and C implementation with Ruby environment dependency | C++ implementation without special requirements |
| Thread support | Support of multithreading | GIL constraint on multithreading | Support of multithreading |
| Hot upgrade | Not supported | Not supported | Supported |
| Centralized configuration management | Not supported | Not supported | Supported |
| Self-detection of the running status | Not supported | Not supported | Support of CPU/memory threshold protection |

Log file collection – Performance comparison

Log sample: The following is a 365-byte Nginx access log with 14 structured fields:

```

42.120.74.166 370261 - [14/Nov/2015:17:50:05 +0800] "POST http://www.xxx.com/auction/order/
unity_order_confirm.htm" 200 1152 "http://www.xxx.com/test_now.jhtml" "Mozilla/5.0 (Windows NT 6.1)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/28.0.1500.72 Safari/537.36" "316312088"
"78c97666dbec0bc3dc5558e4f5a28e55" "ac15399813878147670451784e" center test_local 29374

```

The following test repeatedly writes the log into a file at different simulated pressures. The time field of each log is set to the system time when the log is written, and the other 13 fields are the same for all logs.

The log parsing process in the simulated scenario is the same as that in the actual condition, except that the network traffic generated by the write operation is reduced due to a relatively high data compression rate.

logstash

LogStash 2.0.0 parses logs through the grok and writes parsed logs to Kafka (which has built-in plugs and enables Gzip compression).

Log parsing configuration:

```
grok {
  patterns_dir => "/home/admin/workspace/survey/logstash/patterns"
  match => { "message" => "%{IPORHOST:ip} %{USERNAME:rt} - [%{HTTPDATE:time}] \" %{WORD:method}
%{DATA:url}\" %{NUMBER:status} %{NUMBER:size} \"%{DATA:ref}\" \" %{DATA:agent}\" \" %{DATA:cookie_unb}\"
\" %{DATA:cookie_cookie2}\" \" %{DATA:monitor_traceid}\" \" %{WORD:cell} %{WORD:ups}
%{BASE10NUM:remote_port}\" }
  remove_field => ["message"]
}
```

Test results:

| Write TPS | Write traffic (KB/s) | CPU usage (%) | Memory usage (MB) |
|-----------|----------------------|---------------|-------------------|
| 500 | 178.22 | 22.4 | 427 |
| 1000 | 356.45 | 46.6 | 431 |
| 5000 | 1782.23 | 221.1 | 440 |
| 10000 | 3564.45 | 483.7 | 450 |

fluentd

td-agent-2.2.1 parses logs based on a regular expression and writes parsed logs to Kafka (which has the third-party plug-in fluent-plugin-kafka and 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"(?<cookie_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>
```

Test results:

| Write TPS | Write traffic (KB/s) | CPU usage (%) | Memory usage (MB) |
|-----------|----------------------|---------------|-------------------|
| 500 | 178.22 | 13.5 | 61 |
| 1000 | 356.45 | 23.4 | 61 |
| 5000 | 1782.23 | 94.3 | 103 |

NOTE: A single process of Fluentd uses only one CPU core due to GIL constraints. The multiprocess plug-in can be used to support higher log throughputs.

logtail

Logtail 0.9.4 performs log structuring based on a regular expression and writes LZ4-compressed data to the Log Service over HTTP. batch_size is set to 4,000.

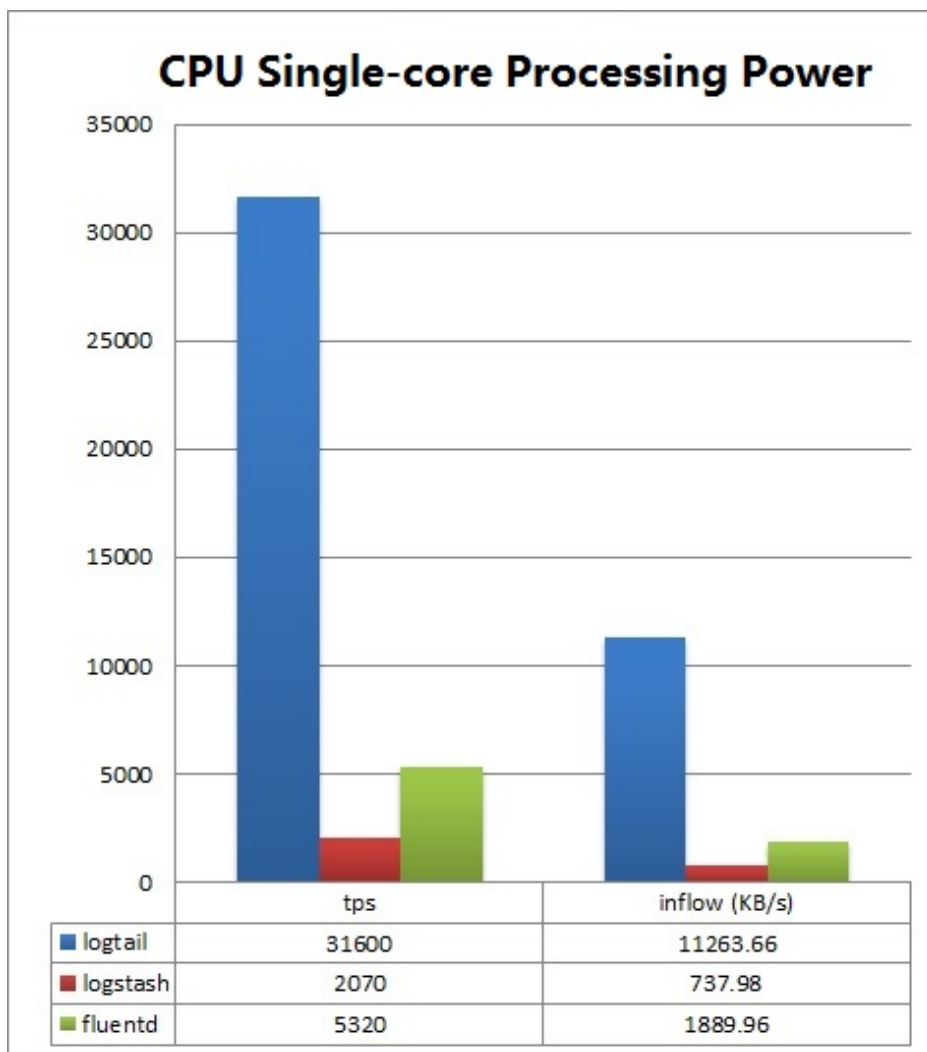
Log parsing configuration:

```
logRegex : (\S+)\s(\d+)\s-
\s\[([^\]]+)\]\s"([^\"]+)\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
```

Test results:

| Write TPS | Write traffic (KB/s) | CPU usage (%) | Memory usage (MB) |
|-----------|----------------------|---------------|-------------------|
| 500 | 178.22 | 1.7 | 13 |
| 1000 | 356.45 | 3 | 15 |
| 5000 | 1782.23 | 15.3 | 23 |
| 10000 | 3564.45 | 31.6 | 25 |

Single-core processing capability comparison



Conclusion

LogStash, Fluentd, and Logtail have the following features:

- LogStash supports common log types, diverse plug-ins, and flexible customization, but it has relatively low performance and is prone to high memory usage due to JVM.
- Fluentd supports common log types and many plug-ins, and it delivers good performance.
- Logtail occupies the least CPU and memory resources of machines, delivers good performance throughput, and provides full support of common log collection scenarios. However, it has no plug-in support and delivers lower flexibility and scalability than LogStash and Fluentd.

Log query

What methods does Log Service provide for you to query collected logs?

Log Service provides three log query methods:

1. Log Service console. For details, see [Query logs](#).
2. SDKs (available in C++, Java, PHP, .Net, and Python). For details, see [SDKs](#).
3. RESTful APIs. For details, see [API](#).

What query capabilities does Log Service provide?

1. Query by a combination of criteria. For details about the query syntax, see [LogSearch syntax](#).
2. Query 10 million logs at a time. Based on the criteria you specified, you can query the logs, read the time-based distribution of matched logs or obtain the raw logs.
3. Caching of queried logs, allowing you to get better results for a second query by the same criteria.

What are the constraints of LogSearch?

1. Log Service supports queries based on a combination of up to 10 keywords.
2. Up to 100 lines of raw data is returned for a single query.
3. Up to 10 million lines of data is processed in a single query.

What if no query results are returned?

1. Go to the **Logstore List** page, find the **Log Consumption** column in **Log Consumption Mode**, click **Preview**, and select **ShardID** to check whether data exists.
 - If data exists, return to the **Query** page of the **Log Index** column. Click the **Topic** text box next to the **Keyword** search box, and check for topics.
 - If topics exist, select a topic and click **Query** to check whether data exists.
 - If neither topics nor data exists after you click **Query**, **open a ticket**.
 - If no data exists, check whether the collected data is complete.

The Log Service provides two functions related to the read operation.

Log collection and consumption (LogHub): provides public channels for log collection and distribution, sequential (FIFO) read and write of full data, and functions similar to Kafka.

- Each LogStore has one or more shards. Data is written to a specific shard at random.

- You can read logs in batches from the specified shard according to the log write sequence.
- You can set the starting point (cursor) of batch log pulling from shards based on the time when the Log Service receives logs.
- By default, logs are retained in LogHub for two days, during which logs can be consumed.

LogSearch (index): The LogSearch function is provided based on LogHub and supports massive log query. Data is queried randomly based on keywords.

- Acquisition of only keyword-matched data
- Boolean combination of keywords AND, NOT, and OR
- Data is queried based on all shards.

Difference:

| Function | LogSearch | Log collection and consumption (based on LogHub) |
|-------------------------------|--|---|
| Keyword search | Supported | Not supported |
| Reading of small data volumes | Fast | Fast |
| Full data reading | Slow (100 logs every 100 ms, not recommended) | Fast (1-MB logs every 10 ms, recommended) |
| Reading by topic | Yes | No. Data is read by shard. |
| Reading by shard | No. Data is queried based on all shards. | Yes. A shard must be specified for the read operation. |
| Fee | Relatively high | Low |
| Application scenario | Data filter is performed for monitoring and troubleshooting. | Full processing scenarios such as stream computing and batch processing |

This document describes common query and analysis errors of Log Service. For details about basic syntaxes, see [Analysis syntax](#).

List of common errors

1. line 1:44: Column 'mykeyfield' cannot be resolved;please add the column in the index attribute
2. Column 'xxxx_line' not in GROUP BY clause;please add the column in the index attribute
3. sql query must follow search query,please read syntax doc
4. key word(where) is not supported,please read query syntax # / select apiName,count() as count where apiName=" " group by apiName order by count desc limit 10
5. please read syntax document,and make sure all related fields are indexed. error after select

.error detail:line 1:10: identifiers must not start with a digit; surround the identifier with double quotes

6. please read syntax document, and make sure all related fields are indexed. error after select
.error detail:line 1:9: extraneous input " " expecting

1. line 1:44: Column 'my_key_field' cannot be resolved; please add the column in the index attribute

Error cause: The key my_key_field does not exist and you cannot reference the key for query.

Solution: On the Query page, add this key as a key value index in the query and analysis attribute at the right-up corner, and enable the statistics function.

2. Column 'xxxxline' not in GROUP BY clause; please add the column in the index attribute

Error cause: The GROUP BY syntax is used for query but a non-agg field which is not contained in GROUP BY is referenced in Select. For example, in select key1, avg(latency) group by key2, key1 is not contained in GROUP BY.

Solution: The correct syntax is select key1, avg(latency) group by key1, key2.

3. sql query must follow search query, please read syntax doc

Error cause: The filter condition is not specified, for example, select ip, count(*) group by ip.

Solution: The correct syntax is *|select ip, count(*) group by ip.

4. key word(where) is not supported, please read query syntax # / select apiName, count() as count where apiName=" " group by apiName order by count desc limit 10

Error cause: The SQL syntax contains a Where condition, which is not allowed.

Solution: Add Where in the filter condition. The correct syntax is apiName:"" | select apiName, count(*) as count group by apiName order by count desc limit 10.

5. please read syntax document, and make sure all related

fields are indexed. error after select .error detail:line 1:10: identifiers must not start with a digit; surround the identifier with double quotes

Error cause: The column name or vector name in SQL starts with a number, which is not allowed.

Solution: The name must start with a letter.

6. please read syntax document,and make sure all related fields are indexed. error after select .error detail:line 1:9: extraneous input " expecting

Error cause: A word spelling is wrong.

Solution: Correct the wrong spelling.

Errors of Log Service:

1. illegal param! [LogContent] is null
2. send data fail, error_code:WriteQuotaExceed error_message:Write quota exceed
projectName:project_name
3. WARN Aliyun.SLS.Logtail.LogFileReader - logfile xxx.log has n old enough logs with first fail case xxx

Solutions

1. illegal param! [LogContent] is null

Check the following configurations:

Check whether you have entered all the required information of the sample log.

Check whether the regular expressions in the first line are correct.

If your issue is still not resolved after the preceding troubleshooting process, send us the sample log and the regular expressions, so that we can reproduce the issue for solutions.

If this issue continues, contact our after-sales technical support.

2. send data fail, error_code:WriteQuotaExceed error_message:Write quota exceed projectName:project_name

If a similar error occurs in `ilogtail.log` when you use Log Service:

send data fail, error_code:WriteQuotaExceed error_message:Write quota exceed
projectName:project_name

there are insufficient quota for writing. The volume of logs you write into it exceeds the predefined threshold, or the write speed is higher than the limit.

Currently, the processing capacity of each shard is:Write: 5 MB/s, 2,000 times/sRead: 10 MB/s, 100 times/s

If your data volume is beyond what the shard can process, you can split the shard. For more information, see [Split the Shard](#).The maximum number of write requests per minute at the project level is 300 thousand. If you write logs using a program, and some requests may exceed the quota limit, we recommend that you write logs in bulk or use `Producer-lib` to limit the maximum packet size for each upload to 3 MB and the maximum entry counts to 4,096.

3. WARN Aliyun.SLS.Logtail.LogFileReader - logfile xxx.log has n old enough logs with first fail case xxx

The following error messages may appear in the `ilogtail.log`:

WARN Aliyun.SLS.Logtail.LogFileReader - logfile xxx.log has n old enough logs with first fail case xxx

This is caused by the old logs.

Log collection rules for Logtail:

Process the historical data separately.

Reduce the cache time for data flushing or even perform real-time flushing.

Pay attention to the time zone when you change the log entries.

Troubleshooting solutions:

When a new log file is monitored by Logtail, the logs written in the log file in the past one minute are considered as old data and discarded.

The new data written in the past five minutes is also considered as old data and discarded. The error indicates that your logs are cached in the memory and has timed out when you actually write them into files.

For the log time beyond the range of -7 days to 360s, such logs are discarded by the server. The error indicates that you do not correctly set the time zone of the log.

If the time zone setting does not conform to rule 1, the historical data is discarded. But if there is no violation of other rules, no error occurs in the subsequent logs.

If rule 2 is violated, an error may occur occasionally, and some logs can be viewed in the console.

If rule 3 is violated, logs are not collected, so they cannot be viewed in the console.

If this issue continues, contact our after-sales technical support.