Log Tank Service

Best Practices

Issue	01
Date	2024-03-05

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Overview

Best practices related to Log Tank Service (LTS):

- **Analyzing Huawei Cloud ELB Access Logs for O&M Insights**
- **Analyzing Huawei Cloud WAF Logs for O&M Insights**
- **Analyzing Application Run Logs (in Log4j Format)**
- **Using Scripts to Invoke LTS APIs for Custom Operations**
- **Collecting Statistics on LTS Expenses of Different Departments Based on Log Stream Tags**

2 Analyzing Huawei Cloud ELB Access Logs for O&M Insights

Introduction

When distributing external traffic, Elastic Load Balance (ELB) logs details of HTTP and HTTPS requests, such as URIs, client IP addresses and ports, and status codes.

You can use ELB access logs for auditing or search for logs by time and keyword. You can also obtain external access statistics by running SQL aggregation queries. For example, you can check the number of requests with 404 responses within a certain day, or analyze the unique visitors (UVs) or page views (PVs) within a week.

Prerequisites

You have and used a Huawei Cloud load balancer.

Restrictions

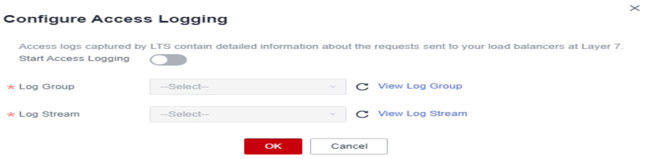
ELB access logs only record layer 7 requests sent to the dedicated and shared load balancers. Layer 4 shared load balancing is not logged.

Procedure

Step 1 Report ELB access logs to LTS. For details, see **Access Logging**.

1. Log in to the management console.
2. Click 📍 in the upper left corner to select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Elastic Load Balance**.
4. On the **Load Balancers** page, click the name of a load balancer.
5. On the **Access Logs** tab, click **Configure Access Log**. Enable access logging, and select an LTS log group and log stream. If necessary, create **a log group** and **a log stream** first.

Figure 2-1 Reporting ELB access logs to LTS



Configure Access Logging

Access logs captured by LTS contain detailed information about the requests sent to your load balancers at Layer 7.

Start Access Logging

* Log Group --Select-- View Log Group

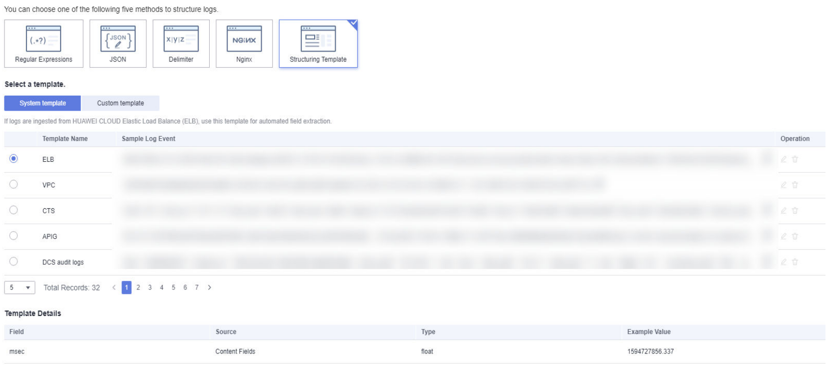
* Log Stream --Select-- View Log Stream

6. Click **OK**.

Step 2 Click ☰ in the upper left corner and choose **Management & Governance** > **Log Tank Service**.

Step 3 On the **Log Management** page, click the target log stream name. On the page that is displayed, click ⚙ in the upper right corner. On the **Cloud Structuring Parsing** tab page, select **Structuring Template** and select the ELB system template for log structuring. You can enable **Quick Analysis** if needed. For details about quick analysis, see **Setting Quick Analysis**.

Figure 2-2 Selecting the ELB structuring template



You can choose one of the following five methods to structure logs.

Select a template.

If logs are ingested from HUAWEI CLOUD Elastic Load Balance (ELB), use this template for automated field extraction.

	Template Name	Sample Log Event	Operation
●	ELB		
○	VPC		
○	CTS		
○	APIG		
○	DCS audit logs		

5 Total Records: 32 < 1 2 3 4 5 6 7 >

Template Details

Field	Source	Type	Example Value
msec	Content Fields	float	1594727856.337

Step 4 On the log stream details page, click **Visualization** and run SQL queries. For details about how to visualize query results, see **Log Structuring**.

- To count the PVs within a week, run the following SQL statement:
```
select count(*) as pv
```
- To count the UVs within a week, run the following SQL statement:
```
select count(distinct remote_port) as uv
```
- Statistics on 2xx/3xx/4xx/5xx (return codes) returned by all URIs in one day are collected to show the service execution result. The SQL query and analysis statements are as follows:
```
select host, router_request_uri as url, count(*) as pv,
sum(case when status >= 200 and status < 300 then 1 else 0 end )  as "2xx times",
sum(case when status >= 300 and status < 400 then 1 else 0 end )  as "3xx times",
sum(case when status >= 400 and status < 500 then 1 else 0 end )  as "4xx times",
sum(case when status >= 500 and status < 600 then 1 else 0 end )  as "5xx times"
group by host, router_request_uri
order by pv desc
limit 100
```
  The query results can be displayed in a table, bar chart, line chart, pie chart, or number chart.

----End

3 Analyzing Huawei Cloud WAF Logs for O&M Insights

Introduction

Web Application Firewall (WAF) examines all HTTP and HTTPS requests to detect and block attacks such as SQL injections, cross-site scripting (XSS), Trojan upload, and command or code injections. You can check the access and attack logs for real-time decision-making, device O&M, and service trend analysis.

Prerequisites

You have purchased and used a Huawei Cloud WAF instance.

Procedure

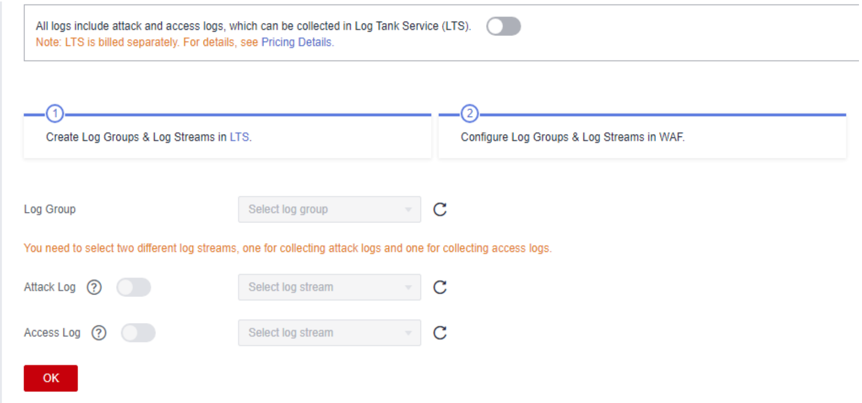
Step 1 Add a website to WAF.

1. Log in to the management console.
2. Click 📍 in the upper left corner to select the desired region and project.
3. Click ☰ in the upper left corner and choose **Security & Compliance** > **Web Application Firewall**.
4. Add the domain name by referring to **Add a Domain Name to WAF**.

Step 2 Enable WAF logging to collect WAF logs to LTS. For details, see **Enabling LTS for WAF Logging**.

1. On the WAF console, choose **Events** in the navigation pane and click the **Configure Logs** tab. Enable logging and select a log group and log stream. If necessary, create **a log group** and **a log stream** first.
2. Click **OK**.

Figure 3-1 Configuring logs



All logs include attack and access logs, which can be collected in Log Tank Service (LTS).

Note: LTS is billed separately. For details, see Pricing Details.

① Create Log Groups & Log Streams in LTS.

② Configure Log Groups & Log Streams in WAF.

Log Group: Select log group

You need to select two different log streams, one for collecting attack logs and one for collecting access logs.

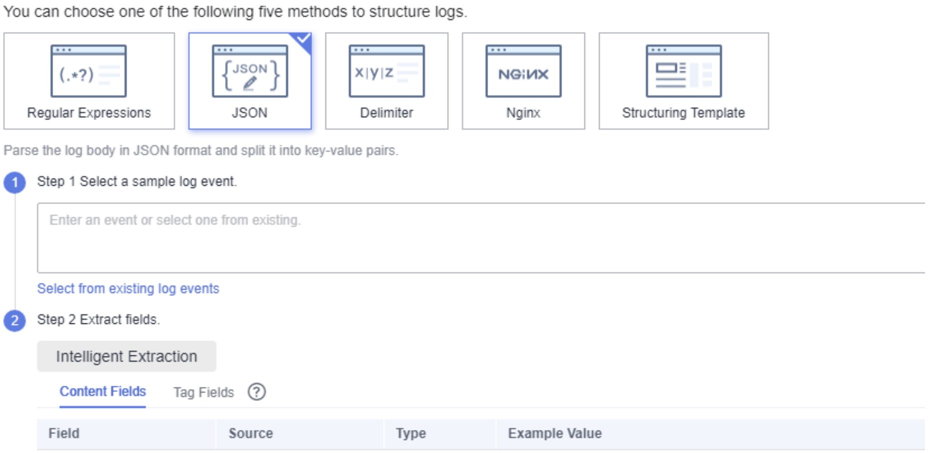
Attack Log: Select log stream

Access Log: Select log stream

Step 3 Click ☰ in the upper left corner and choose **Management & Governance** > **Log Tank Service**.

Step 4 On the **Log Management** page, click the name of a log stream to access its details page, click ⚙ in the upper right corner. Click the **Cloud Structuring Parsing** tab. Select **JSON**, select a sample log event, and complete the configuration. For details, see **Log Structuring**.

Figure 3-2 Configuring logs in JSON format



You can choose one of the following five methods to structure logs.

Parse the log body in JSON format and split it into key-value pairs.

Step 1 Select a sample log event.

Enter an event or select one from existing.

Select from existing log events

Step 2 Extract fields.

Field	Source	Type	Example Value

Step 5 On the log stream details page, click **Visualization** and run SQL queries. For details about how to visualize query results, see **Log Structuring**.

- To count the number of attacks within a week, run the following SQL statement:

```
select count(*) as attack_times
```

- To count the number of attacks by type in one day, run the following SQL statement:

```
select attack,count(*) as times group by attack
```

  The query results can be displayed in a table, bar chart, line chart, pie chart, or number chart.

----End

4 Analyzing Application Run Logs (in Log4j Format)

Introduction

Log4j is Apache's open-source project used for logging. We can calculate the number and proportion of logs at different levels, or gather statistics on services from run logs.

For example, you can know the transaction volume of an offering on a day from logs such as the following:

```
2020-12-28_21:10:48.081 [http-nio-8083-exec-6] INFO  discounted shoes - num is :9
```

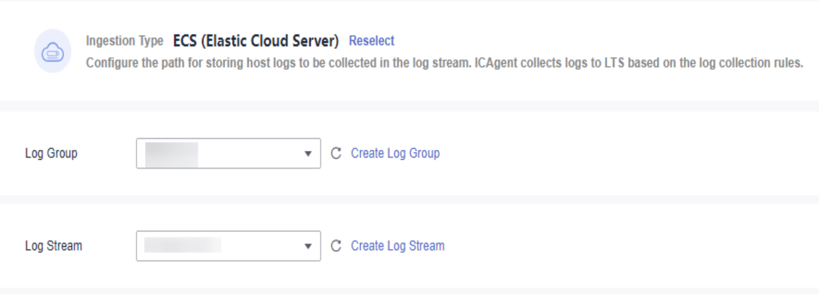
Procedure

Step 1 Log in to the LTS console and choose **Log Ingestion** in the navigation pane.

Step 2 Click **Elastic Cloud Server (ECS)** to configure log ingestion.

Step 3 Select a log stream.

1. Select a log group from the drop-down list of **Log Group**. If there are no desired log groups, click **Create Log Group** to create one.
2. Select a log stream from the drop-down list of **Log Stream**. If there are no desired log streams, click **Create Log Stream** to create one.
3. Click **Next: (Optional) Select Host Group**.



Ingestion Type **ECS (Elastic Cloud Server)** Reselect

Configure the path for storing host logs to be collected in the log stream. ICAgent collects logs to LTS based on the log collection rules.

Step 4 Select host groups.

1. Select one or more host groups from which you want to collect logs. If there are no desired host groups, click **Create** above the host group list to create one. For details, see **Managing Host Groups**.

   NOTE

   You can choose not to select a host group in this step, but associate a host group with the ingestion configuration after you finish the procedure here. To do this, either:

   - Choose **Host Management** in the navigation pane, click the **Host Groups** tab, and make the association, or
   - Choose **Log Ingestion** in the navigation pane, click an ingestion configuration, and make the association on the details page.

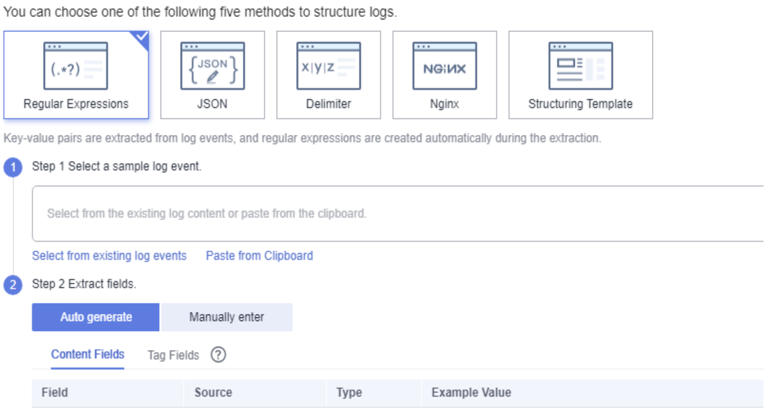
2. Click **Next: Collection Configuration**.

Step 5 Configure the collection.

1. Configure the collection parameters. For details, see **Configuring Collection**.
2. Click **Submit**.

Step 6 On the log stream details page, click ⚙ . On the **Cloud Structuring Parsing** page, select **Regular Expressions**, select a log event, and extract four fields: **Time1**, **ThreadName**, **Level**, and **Message**, as shown in **Figure 4-1**.

Figure 4-1 Structuring logs with regular expressions



You can choose one of the following five methods to structure logs.

Key-value pairs are extracted from log events, and regular expressions are created automatically during the extraction.

1 Step 1 Select a sample log event.

Select from the existing log content or paste from the clipboard.

2 Step 2 Extract fields.

Field	Source	Type	Example Value

Step 7 On the log stream details page, click **Visualization** and run SQL queries. For details about how to visualize query results, see **Log Structuring**.

- To query the error type distribution in the last seven days, run the following SQL statement:

  ```
  SELECT Level, count(*) as Number group by Level
  ```

- To query the running threads in the last 5 minutes, set the time range to **Last 5 minutes** and run the following SQL statement:

  ```
  SELECT distinct(ThreadName)
  ```

- To query the total transaction volume of a product, run the following SQL statement:

  ```
  SELECT sum(cast(regexp_extract(Message, 'num is\s:(?<Total>[\d]+)', 1) as double)) as Total WHERE
  Message like '%shoes%'
  ```

----End

5 Using Scripts to Invoke LTS APIs for Custom Operations

Introduction

Some configuration-related operations on the LTS console, such as ingesting logs, creating alarms, and configuring log transfer, need to be performed repeatedly. However, the LTS console does not support batch operations. In this case, you can use Python scripts and LTS APIs to perform custom batch operations.

Scenario

- If you have created 1000 rules for log transfer to OBS but set the file time zone to UTC during the transfer, you need to change the time zone to UTC +08:00.
- Currently, the LTS console does not allow batch modifying log transfer rules. Manually modifying each transfer rule will be time-consuming.

Prerequisites

1. **Linux**
2. **Querying API-related documents**
   - Obtain information about all transfer tasks by calling the log transfer API.
   - Change the time zone configured for the transfer task by updating the log transfer API.
3. **Testing API functions in API Explorer**

   API Explorer provides the API search and platform debugging capabilities.
4. **Installing the Python SDK by referring to the API Explorer sample code**
   - Python **SDK dependency address** and **SDK usage description**
     ```
     pip install huaweicloudsdklts
     ```
   - API Explorer provides sample code for calling APIs using Python. The following is an example:
     ```
     # coding: utf-8
     from huaweicloudsdkcore.auth.credentials import BasicCredentials
     from huaweicloudsdklts.v2.region.lts_region import LtsRegion
     from huaweicloudsdkcore.exceptions import exceptions
     from huaweicloudsdklts.v2 import *
     ```

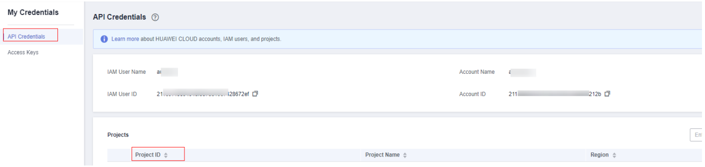
```
if __name__ == "__main__":
    AK = "your ak"
    SK = "your sk"
    PROJECT_ID = "your project id"
    REGION = "your region"
    IAM_ENDPOINT = "iam_endpoint"

    credentials = BasicCredentials(AK, SK,
PROJECT_ID).with_iam_endpoint(IAM_ENDPOINT)
    client = LtsClient.new_builder()
    .with_credentials(credentials)
    .with_region(LtsRegion.value_of(REGION))
    .build()

    try:
        request = ListTransfersRequest()
        request.log_transfer_type = "OBS"
        response = client.list_transfers(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Procedure

Step 1 Obtain the parameter and replace it with the actual value in the code.

- **Obtain an AK/SK Pair**
- Obtain the project ID. For details, see **API Credentials**.

Figure 5-1 Obtaining the project ID



IAM User Name ai[redacted]

Account Name a[redacted]

IAM User ID 21[redacted]28672ef

Account ID 211[redacted]212b

- Obtain the value of **Region&iam_Endpoint** from **Regions and Endpoints**.

Table 5-1 Endpoints

Region Name	Region	Endpoint	Protocol
AP-Bangkok	ap-southeast-2	lts.ap-southeast-2.myhuaweicloud.com	HTTPS
AP-Singapore	ap-southeast-3	lts.ap-southeast-3.myhuaweicloud.com	HTTPS
CN-Hong Kong	ap-southeast-1	lts.ap-southeast-1.myhuaweicloud.com	HTTPS

- **Time Zone and Time Zone ID**

Table 5-2 Common time zones

Time Zone	Time Zone ID
UTC-12:00	Etc/GMT+12
UTC-11:00	Etc/GMT+11
UTC-10:00	Pacific/Honolulu
UTC-09:00	America/Anchorage
UTC-08:00	America/Santa_Isabel
UTC-07:00	America/Chihuahua
UTC-06:00	America/Chicago
UTC-05:00	America/New_York
UTC-04:00	America/Santiago
UTC-03:00	America/Sao_Paulo
UTC-02:00	Etc/GMT+2
UTC-01:00	Atlantic/Azoresjavik
UTC+00:00	Europe/London
UTC+01:00	Europe/Parist
UTC+02:00	Europe/Istanbul
UTC+03:00	Europe/Minsk
UTC+04:00	Europe/Moscow
UTC+05:00	Asia/Tashkent
UTC+06:00	Asia/Almaty
UTC+07:00	Asia/Bangkok
UTC+08:00	Asia/Shanghai
UTC+09:00	Asia/Tokyo
UTC+10:00	Asia/Yakutsk
UTC+11:00	Asia/Vladivostok
UTC+12:00	Pacific/Fiji
UTC+13:00	Pacific/Apia

Step 2 Check whether the **huaweicloudsdkcore** and **huaweicloudsdklts** packages have been installed.

```
pip list | grep huaweicloudsdk
```

If not, perform the following operations:

```
pip install huaweicloudsdkcore huaweicloudsdklts
```

Step 3 Create a file named **lts_python.py** and copy the following code to the file:

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdklts.v2 import *
from huaweicloudsdklts.v2.region.lts_region import LtsRegion

if __name__ == "__main__":
AK = "your ak"
SK = "your sk"
PROJECT_ID = "your project id"
REGION = "your region"
IAM_ENDPOINT = "iam_endpoint"

OBS_TIME_ZONE = "the time_zone you want to change"
OBS_TIME_ZONE_ID = "time_zone_id"

credentials = BasicCredentials(AK, SK, PROJECT_ID).with_iam_endpoint(IAM_ENDPOINT)

client = LtsClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(LtsRegion.value_of(REGION)) \
    .build()

# 1.get obs transfer task
try:
    request = ListTransfersRequest()
    request.log_transfer_type = "OBS"
    response = client.list_transfers(request)
    obs_transfer_num = len(response.log_transfers)
    task_list = response.log_transfers
    print("#### get {} obs transfer task ####".format(obs_transfer_num))
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)

# 2.set obs transfer task obs_time_zone to UTC+08:00
CNT = 1
while len(task_list):
    transfer_task = task_list.pop()
    print("There are still {} progress: \n".format(len(task_list)), transfer_task)
    try:
        if transfer_task.log_transfer_info.log_transfer_detail.obs_time_zone == OBS_TIME_ZONE:
            CNT += 1
            continue
        request = UpdateTransferRequest()
        transfer_task.log_transfer_info.log_transfer_detail.obs_time_zone = OBS_TIME_ZONE
        transfer_task.log_transfer_info.log_transfer_detail.obs_time_zone_id = OBS_TIME_ZONE_ID
        request.body = UpdateTransferRequestBody(
            log_transfer_info=transfer_task.log_transfer_info,
            log_transfer_id=transfer_task.log_transfer_id
        )
        response = client.update_transfer(request)
        CNT += 1
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
        task_list.append(transfer_task)
    except exceptions.ServerResponseException as e:
        print({
            "target": transfer_task.log_streams,
            "reason": e
        })
        task_list.append(transfer_task)
```

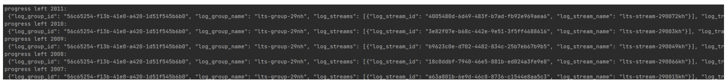
Step 4 Run the Python script to change the time zone of log transfer to OBS in batches.

```
nohup python lts_python.py > lts_python.log &
```

View execution logs.

```
tail -f lts_python.log
```

Figure 5-2 Viewing execution logs



```
progress left 2011:
 {"log_group_id": "56c65254-f13b-41e0-a420-1d51f545b6b0", "log_group_name": "lts-group-29nh", "log_streams": [{"log_stream_id": "4005480d-6d49-483f-b7ad-fb92e969aea6", "log_stream_name": "lts-stream-290072kh"}], "log_tr
progress left 2010:
 {"log_group_id": "56c65254-f13b-41e0-a420-1d51f545b6b0", "log_group_name": "lts-group-29nh", "log_streams": [{"log_stream_id": "3e82f07e-b68c-442e-9e51-3f5ff4688616", "log_stream_name": "lts-stream-29003kh"}], "log_tra
progress left 2009:
 {"log_group_id": "56c65254-f13b-41e0-a420-1d51f545b6b0", "log_group_name": "lts-group-29nh", "log_streams": [{"log_stream_id": "b9623c0e-d702-4482-834c-25b7eb67b9b5", "log_stream_name": "lts-stream-290049kh"}], "log_tr
progress left 2008:
 {"log_group_id": "56c65254-f13b-41e0-a420-1d51f545b6b0", "log_group_name": "lts-group-29nh", "log_streams": [{"log_stream_id": "18c0ddbf-7940-46e5-881b-ed024a3fe9e8", "log_stream_name": "lts-stream-290066kh"}], "log_tr
progress left 2007:
 {"log_group_id": "56c65254-f13b-41e0-a420-1d51f545b6b0", "log_group_name": "lts-group-29nh", "log_streams": [{"log_stream_id": "a63a801b-6e9d-46c8-8736-c1544e8aa5c3", "log_stream_name": "lts-stream-290015kh"}], "log_tr
```

----End

6 Collecting Statistics on LTS Expenses of Different Departments Based on Log Stream Tags

To collect statistics on the expenses of different departments in an enterprise on LTS, you can add tags to LTS log streams to identify different service departments. LTS will add these tags when uploading CDRs to the billing center. You can download LTS billing information from **Billing** > **Bills** > **Bill Details**. Then, you can collect statistics on expenses of different departments based on resource tags, providing a basis for expense allocation of the enterprise.

Prerequisites

The function of reporting CDRs by log stream is being tested by friendly users. To use this function to collect statistics on the expenses of different departments in LTS, **submit a service ticket**.

Solution

Collect statistics on LTS expenses of different departments by adding log stream tags.

This practice uses departments aa and bb as an example. First, add the group=groupaa tag to the log group of department aa and the group=groupbb tag to the log group of department bb. Then, export bills from the fee details and perform statistical analysis with Excel.

NOTE

The prices mentioned in the following are only for reference. For details, see **Price Calculator**.

Procedure

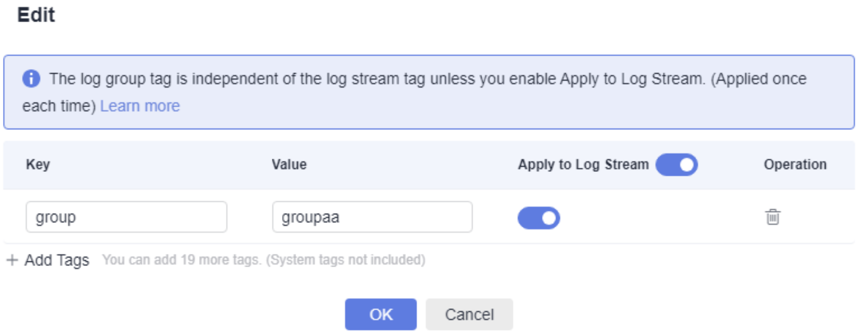
Step 1 Log in to the LTS console.

Step 2 Move the cursor to the **Tag** column of the target log group and click ✎.

NOTE

By adding a tag to a log group, you can apply the tag to an existing log stream in the log group. If a new log stream is added to the log group, you need to add a tag to the log stream separately. For details, see **Tagging a Log Stream**.

Step 3 On the **Edit** dialog box displayed, click **Add Tags**, enter the tag key **group** and tag value **groupaa** of department aa, and click **OK**.



Edit

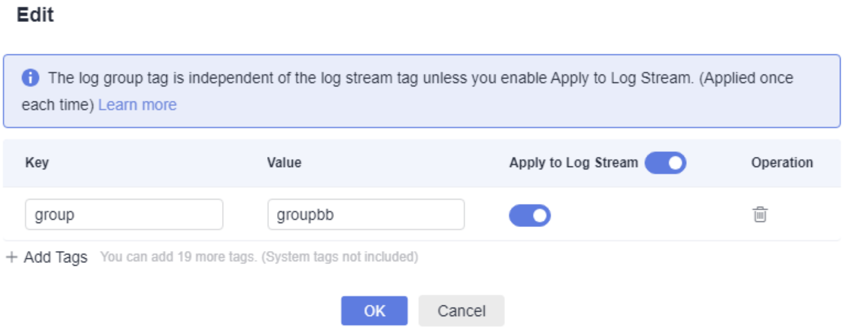
The log group tag is independent of the log stream tag unless you enable Apply to Log Stream. (Applied once each time) Learn more

Key	Value	Apply to Log Stream ●	Operation
group	groupaa	●	🗑

+ Add Tags You can add 19 more tags. (System tags not included)

Step 4 Move the cursor to the **Tag** column of the target log group and click ✎.

Step 5 On the **Edit** dialog box displayed, click **Add Tags**, enter the tag key **group** and tag value **groupbb** of department bb, and click **OK**.



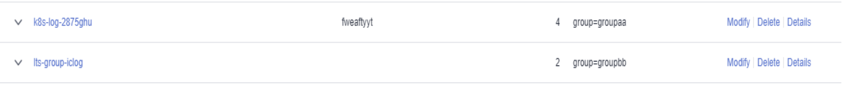
Edit

The log group tag is independent of the log stream tag unless you enable Apply to Log Stream. (Applied once each time) Learn more

Key	Value	Apply to Log Stream ●	Operation
group	groupbb	●	🗑

+ Add Tags You can add 19 more tags. (System tags not included)

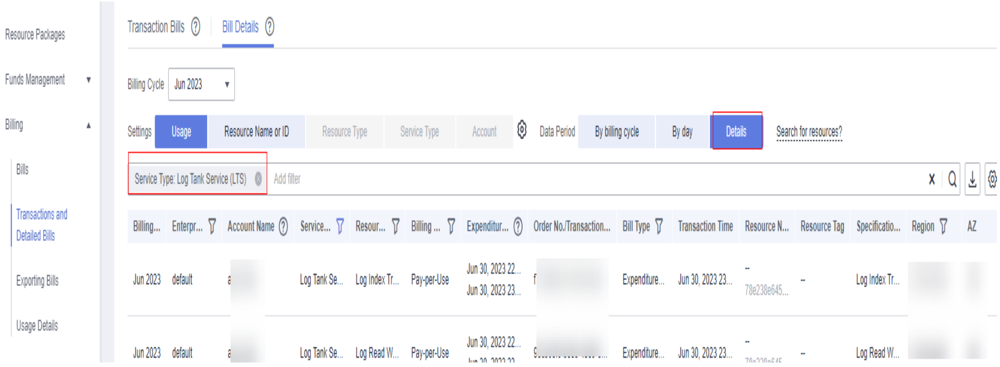
Step 6 After the tag is added successfully, the following information is displayed. It is estimated that the CDR can be generated one hour later. The screenshot is for reference only.



k8s-log-2875ghu	fweaftyyt	4	group=groupaa	Modify | Delete | Details
lts-group-iclog		2	group=groupbb	Modify | Delete | Details

Step 7 On the top menu bar, choose **Billing & Costs** > **Bills**. The **Bills** page is displayed.

Step 8 Choose **Transactions and Detailed Bills** > **Bill Details**, choose a billing cycle, select **Details** for **Data Period**, and choose **Service Type: Log Tank Service (LTS)** for filters.



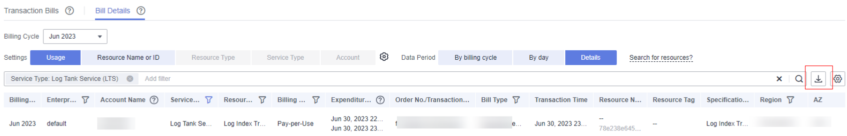
Billing Cycle: Jun 2023

Settings: Usage | Data Period: Details

Service Type: Log Tank Service (LTS)

Billing...	Enterpr...	Account Name	Service...	Resour...	Billing ...	Expenditur...	Order No./Transaction...	Bill Type	Transaction Time	Resource N...	Resource Tag	Specificatio...	Region	AZ
Jun 2023	default	a	Log Tank Se...	Log Index Tr...	Pay-per-Use	Jun 30, 2023 22... Jun 30, 2023 23...	f	Expenditure...	Jun 30, 2023 23...	-- 78e238e645...	--	Log Index Tr...		
Jun 2023	default	a	Log Tank Se...	Log Read W...	Pay-per-Use	Jun 30, 2023 22...	9	Expenditure...	Jun 30, 2023 23...	--	--	Log Read W...		

Step 9 Click ⤓ to export fee details to the local host.



Billing Cycle: Jun 2023

Settings: Usage | Data Period: Details

Service Type: Log Tank Service (LTS)

Billing...	Enterpr...	Account Name	Service...	Resour...	Billing ...	Expenditur...	Order No./Transaction...	Bill Type	Transaction Time	Resource N...	Resource Tag	Specificatio...	Region	AZ
Jun 2023	default		Log Tank Se...	Log Index Tr...	Pay-per-Use	Jun 30, 2023 22... Jun 30, 2023 23...	f	e...	Jun 30, 2023 23...	-- 78e238e645...	--	Log Index Tr...		

Step 10 In the exported excel file, filter **Resource Tag** column to view the expenditure details of departments aa and bb.

NOTE

The prices mentioned in the figure are only for reference. For details, see **Price Calculator**.

----End