Cloud Log Service

Developer Guide

Product Documentation

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

Developer Guide

Embedding CLS Console

Last updated：2024-01-20 17:09:26

Use Cases

CLS allows you to embed the CLS console into an external system, so that you can conduct log search and analysis without logging in to the Tencent Cloud console. This feature offers the following benefits:

Quickly integrate CLS search and analysis capabilities into an external service system (e.g., for business maintenance or operation).

Easily share your log data with others without needing to manage additional Tencent Cloud sub-accounts.

Demo Code for Login-Free Implementation

Directions

1. On the CAM page, create a CAM role, set the role entity to **Tencent Cloud Account**, and select **Allow the current role to access console**. Then, configure the target access permission for the CAM role, for example, the read-only policy permission `QcloudCLSReadOnlyAccess` , name it `CLSReadOnly` , and copy its `RoleArn` information.
2. On the **Policies** page, create a custom policy and select **Create by Policy Generator**. Then, select the JSON tag and enter the following information in **Policy Content**. Note that you need to replace `${YOUR_UIN}` with the account UIN (the resource content is the `RoleArn` of the created role; modify the policy name in case of any inconsistency). Click **Next** and set **Policy Name** to `PlayClsPolicy` .

```
{
    "version": "2.0",
    "statement": [
        {
            "effect": "allow",
            "action":[
                "sts:AssumeRole"
            ],
            "resource": [
                "qcs::cam::uin/${YOUR_UIN}:roleName/CLSReadOnly"
            ]
            }
        ]
    }
```

3. On the **Create User** page, select **Custom Creation** and set **Type** to **Access Resources and Receive Messages**, **Username** to **PlayClsUser**, **Access Method** to **Programming access**, and **User permissions** to `PlayClsPolicy` created in the previous step. After submitting the user creation operation, copy the generated `SecretId` and `SecretKey` .

4. Clone the **demo code**cls-iframe-demo for login-free implementation. As instructed in the `ReadMe` content of the demo project, create the `.env` file in the root directory and enter the required parameters `RoleArn` , `SecretId` , and `SecretKey` .

Note:

Code leakage may lead to the leakage of `SecretId` and `SecretKey` , thereby affecting your account security. We recommend that you use a key in a more secure manner as instructed in the TencentCloud API key security solution and use a sub-account key with the least privilege.

Check the effect after running the project as instructed in the `ReadMe` document of the demo program for login-free implementation.

Note:

This example does not include the authentication logic of external systems. After deployment, all users (even if they have not logged in to Tencent cloud) can view the data under their accounts with the role permissions configured in the example. To ensure data privacy and security, add the authentication logic of external systems or restrict their access to the private network only to ensure that only authorized users can view the page.

5. Concatenate the destination login-free address `s_url` of CLS (optional). If you enter the obtained address in the configuration file of the login-free project, access to the login-free service will be automatically redirected to this address.

The basic address of the CLS search and analysis page:

```
https://console.intl.cloud.tencent.com/cls/search?region=<region>&topic_id=<topic_i
```

Parameters in the CLS search and analysis page URL:

Parameter	Required	Type	Description
region	Yes	String	Region abbreviation, e.g., ap-shanghai for Shanghai region. For other available region abbreviations, see Available Regions
topic_id	No	String	Log topic ID

logset_name	No	String	Logset name
topic_name	No	String	Log topic name
time	No	String	Time range for log search. Format example:2021-07-15T10:00:00.000,2021-07-15T12:30:00.000
queryBase64	No	String	Search and analysis statement, which is base64Url-encoded
hideWidget	No	Boolean	Indicates whether to hide agent/documentation button in the bottom-right corner. `true`: Yes; `false`: No (default)
hideTopNav	No	Boolean	Indicates whether to hide the top navigation bar in the Tencent Cloud console. `true`: Yes; `false`: No (default)
hideLeftNav	No	Boolean	Indicates whether to hide the left navigation bar in the Tencent Cloud console. `true`: Yes; `false`: No (default)
hideTopicSelect	No	Boolean	Indicates whether to hide the log topic selection controls (including the region, logset, and log topic controls). `true`: Yes; `false`: No (default)
hideHeader	No	Boolean	Indicates whether to hide the log topic selection control and the row where the control resides. `true`: Yes; `false`: No (default). This parameter is valid only when `hideTopicSelect` is `true`.
hideTopTips	No	Boolean	Indicates whether to hide the announcements on the top of the page. `true`: Yes; `false`: No (default)
hideConfigMenu	No	Boolean	Indicates whether to hide the log topic configuration management menu. `true`: Yes; `false`: No (default)
hideLogDownload	No	Boolean	Indicates whether to hide the raw log download button. `true`: Yes; `false`: No (default)

Note:

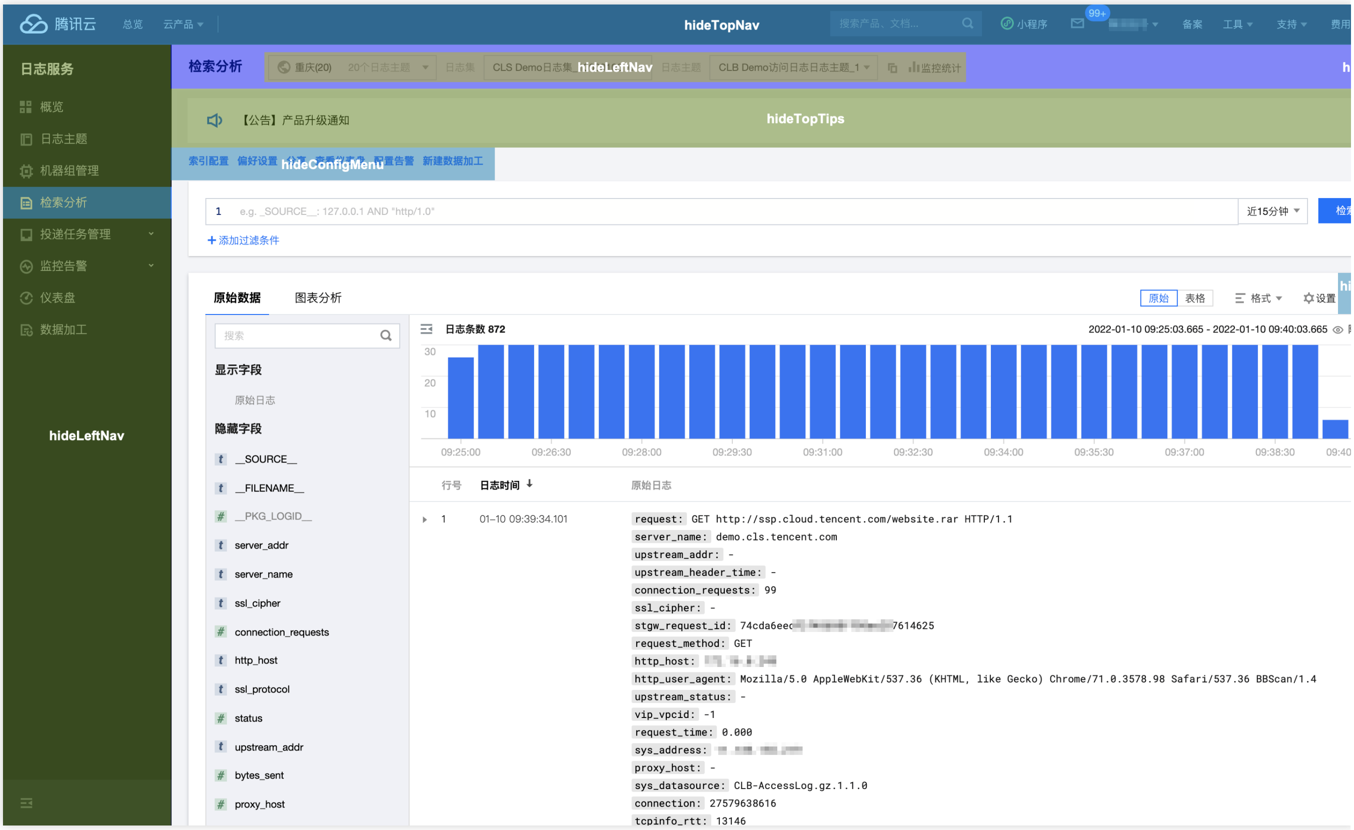
You can specify the log topic to search using URL parameters in either the following modes:

topic_id: use the log topic ID to specify the log topic to search.

logset_name+topic_name: use the logset name and log topic name to specify the log topic to search. Note that if the logset or log topic name changes, the URL adopting this mode will become invalid.

If `topic_id`, `logset_name`, and `topic_name` exist at the same time, `topic_id` prevails.

Relationship between hidden parameters and page modules:



```
1    01-10 09:39:34.101    request: GET http://ssp.cloud.tencent.com/website.rar HTTP/1.1
                           server_name: demo.cls.tencent.com
                           upstream_addr: -
                           upstream_header_time: -
                           connection_requests: 99
                           ssl_cipher: -
                           stgw_request_id: 74cda6eec[redacted]7614625
                           request_method: GET
                           http_host: [redacted]
                           http_user_agent: Mozilla/5.0 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36 BBScan/1.4
                           upstream_status: -
                           vip_vpcid: -1
                           request_time: 0.000
                           sys_address: [redacted]
                           proxy_host: -
                           sys_datasource: CLB-AccessLog.gz.1.1.0
                           connection: 27579638616
                           tcpinfo_rtt: 13146
```

Self-Development for Login-Free Implementation

Directions

Note:

Note: Code leakage may lead to the leakage of `SecretId` and `SecretKey` , thereby affecting your account security. We recommend that you use a key in a more secure manner as instructed in the TencentCloud API key security solution and use a sub-account key with the least privilege.

1. Configure the CLS read-only role, custom policy of the target role, and sub-account bound to the custom policy as instructed in **Demo Code for Login-Free Implementation**. Then, save the `RoleArn` , `SecretId` , and `SecretKey` information.

2. Get the destination login-free address `s_url` as needed as instructed in **Demo Code for Login-Free Implementation**.

3. Repeat the following steps every time you need to open a login-free page.

4. Call the STS AssumeRole API with the obtained key to apply for the temporary key of the target role.

5. Generate the login signature information with the obtained temporary key.

5.1 Sort parameters to be signed.

Sort parameters to be signed listed below in ascending alphabetical or numerical order. That is, sort the parameters by their first letters, then by their second letters if their first letters are the same, and so on. You can do this with the aid of sorting functions in programming languages, such as the ksort function in PHP.

Parameter	Required	Type	Description
action	Yes	String	Action; fixed as `roleLogin`
timestamp	Yes	Int	Current timestamp
nonce	Yes	Int	Random integer. Value range: 10000-100000000
secretId	Yes	String	Temporary AK returned by STS

5.2 Combine the parameters.

Combine the above sorted parameters into the form of "parameter name=parameter value". Example:

```
action=roleLogin&nonce=67439&secretId=AKI***PLE&timestamp=1484793352
```

5.3 Concatenate a signature string.

Construct a signature string in the format of “request method + request CVM + request path + ? + request string”.

Parameter	Required	Description
Request CVM and path	Yes	Fixed as cloud.tencent.com/login/roleAccessCallback
Request method	Yes	GET or POST

Sample signature string

```
GETcloud.tencent.com/login/roleAccessCallback?action=roleLogin&nonce=67439&secretId
```

5.4 Generate a signature string.

Currently, you can sign a string using HMAC-SHA1 or HMAC-SHA256. The sample code in PHP is as follows:

```
$secretKey = 'Gu5***1qA';
$srcStr    = 'GETcloud.tencent.com/login/roleAccessCallback?action=roleLogin&nonce=
$signStr   = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

Sample code for PHP

```
$secretId  = "AKI***";              //Temporary AK returned by STS
$secretKey = "Gu5***PLE";           //Temporary SecretKey returned by STS
$token     = "ADE***fds";           //Security Token returned by STS
$param["nonce"]     = 11886;        //rand(10000,100000000);
$param["timestamp"] = 1465185768; //time();
$param["secretId"]  = $secretId;
$param["action"]    = "roleLogin";
ksort($param);
$signStr = "GETcloud.tencent.com/login/roleAccessCallback?";
foreach ( $param as $key => $value ) {
    $signStr = $signStr . $key . "=" . $value . "&";
}
$signStr   = substr($signStr, 0, -1);
$signature = base64_encode(hash_hmac("sha1", $signStr, $secretKey, true));
echo $signature.PHP_EOL;
```

6. Combine your login information and destination page URL into a login URL.

Parameter values need to be URL-encoded.

```
https://cloud.tencent.com/login/roleAccessCallback?
algorithm=<Encryption algorithm for signing. Currently, only `sha1` and `sha256` ar
&secretId=<secretId for signing>
```

```
&token=<Temporary key token>
&nonce=<nonce for signing>
&timestamp=<Timestamp for signing>
&signature=<Signature string>
&s_url=<Destination URL after login>
```

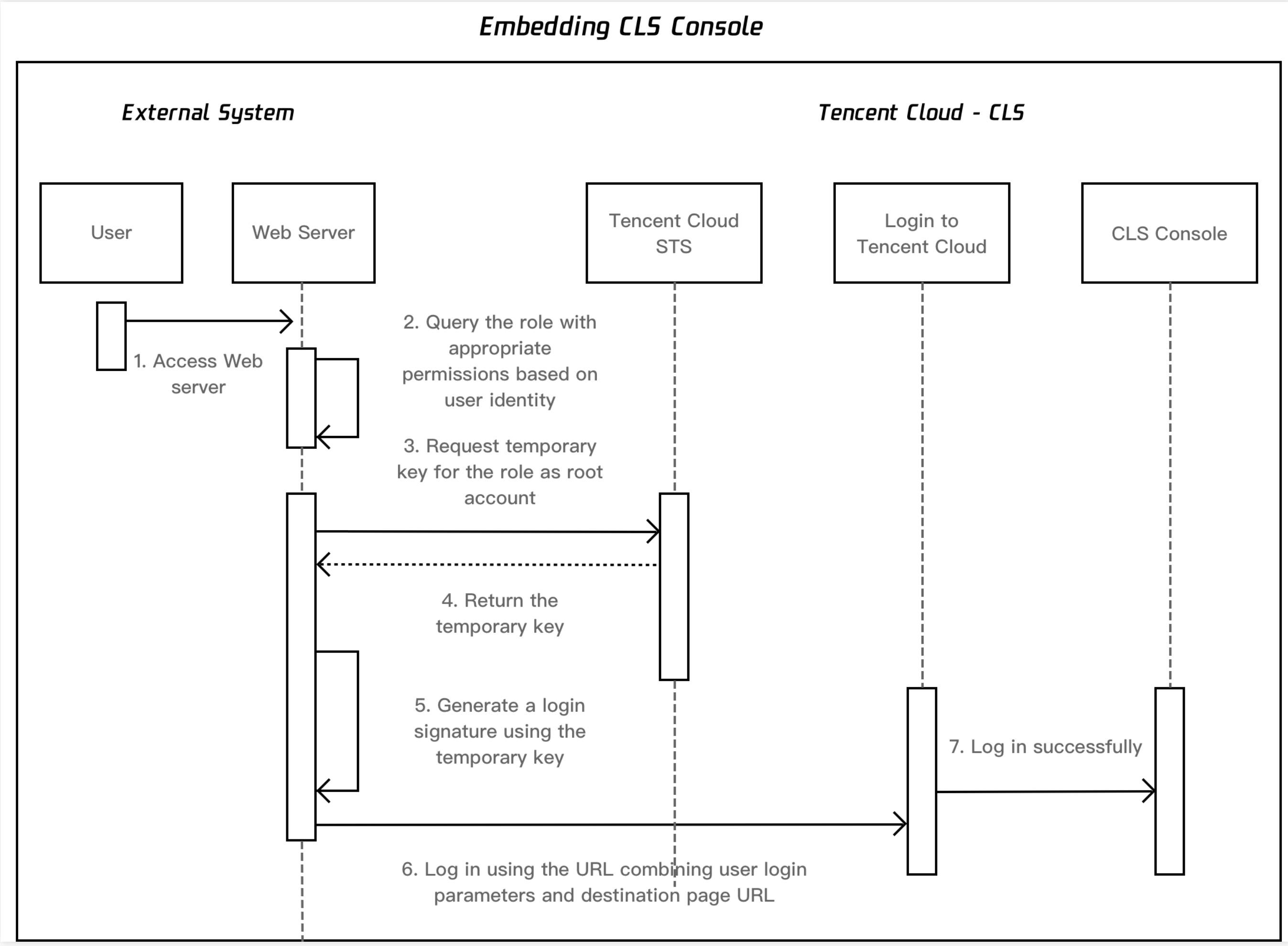
7. Use the final URL to access the embedded CLS page of the Tencent Cloud console. The sample below is a URL to the CLS search analysis page:

```
https://cloud.tencent.com/login/roleAccessCallback?nonce=52055817&s_url=https%3A%2F
```

How It Works

The login-free solution is implemented based on STS.

The login flowchart is as shown below:


Embedding CLS Console
External System
Tencent Cloud - CLS
User
Web Server
Tencent Cloud STS
Login to Tencent Cloud
CLS Console
1. Access Web server
2. Query the role with appropriate permissions based on user identity
3. Request temporary key for the role as root account
4. Return the temporary key
5. Generate a login signature using the temporary key
6. Log in using the URL combining user login parameters and destination page URL
7. Log in successfully

CLS Connection to Grafana

Last updated：2024-01-20 17:08:20

Overview

CLS can be connected to Grafana to export the raw log data and SQL aggregate analysis results for display in Grafana. To display the results in Grafana, you only need to install the Tencent Cloud Monitor Grafana App and enter search and analysis statements in Grafana.

This document describes how to install and configure Grafana on CentOS.

Directions

Installing Grafana

1. For more information on how to install Grafana 8.0 or later, see Install Grafana.

If your Grafana version is earlier than 8.0, configuration backup and upgrade are required. For details, see Upgrade Grafana.

The following example shows how to install Grafana 8.4.3 (click here to get the latest version address) on CentOS:

```
sudo yum install initscripts urw-fonts wget
wget https://dl.grafana.com/oss/release/grafana-8.4.3-1.x86_64.rpm
sudo yum install grafana-8.4.3-1.x86_64.rpm
sudo systemctl daemon-reload
sudo systemctl start grafana-server
sudo systemctl status grafana-server
sudo systemctl enable grafana-server
```

2. After installation, we recommend you modify the [dataproxy] configuration in the `grafana.ini` file.

The default timeout period of Grafana is 30s. When you search for a large amount of data, a timeout may occur. For more information, see Grafana proxy queries timeout after 30s with dataproxy.timeout and dataproxy.keep_alive_seconds to > 30s #35505. We recommend you set the timeout period to 60s to maximize the CLS capabilities. Modify the configuration as follows:

```
[dataproxy]
timeout = 60
dialTimeout = 60
keep_alive_seconds = 60
```

To install more visual panels (such as pie and graph panels), install Grafana panel plugins by running the corresponding commands.

For example, if you want to install the pie panel, you can run the following command:

```
grafana-cli plugins install grafana-piechart-panel
service grafana-server restart
```

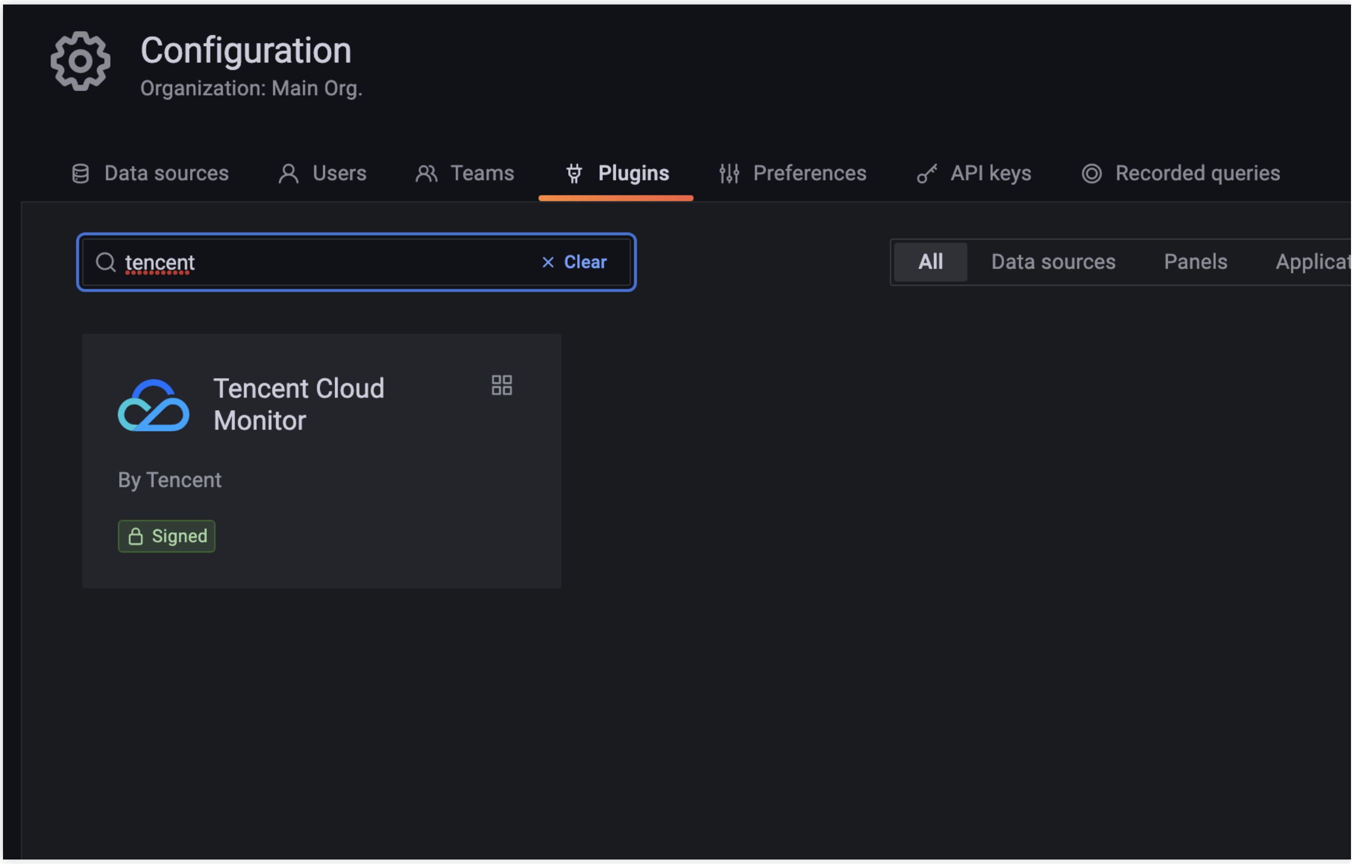
For information on more plugins, see Grafana Plugins.

Installing Tencent Cloud Monitor Grafana App

Installation from the official plugin library

1. Go to the Grafana page.

2. On the **Configuration** > **Plugin** page, search for **Tencent Cloud Monitor** and click it to install it.



Installation on the command line

1. Install Tencent Cloud Monitor Grafana App.

```
grafana-cli plugins install tencentcloud-monitor-app
# If the plugin cannot be found after the installation, it may be because that the
# If your CVM instance is not on CentOS, confirm the location of the Grafana plugin
grafana-cli --pluginsDir ./  plugins install tencentcloud-monitor-app

# If you want to install an unofficial latest version of the plugin or install a be
grafana-cli --pluginUrl https://github.com/TencentCloud/tencentcloud-monitor-grafan
```

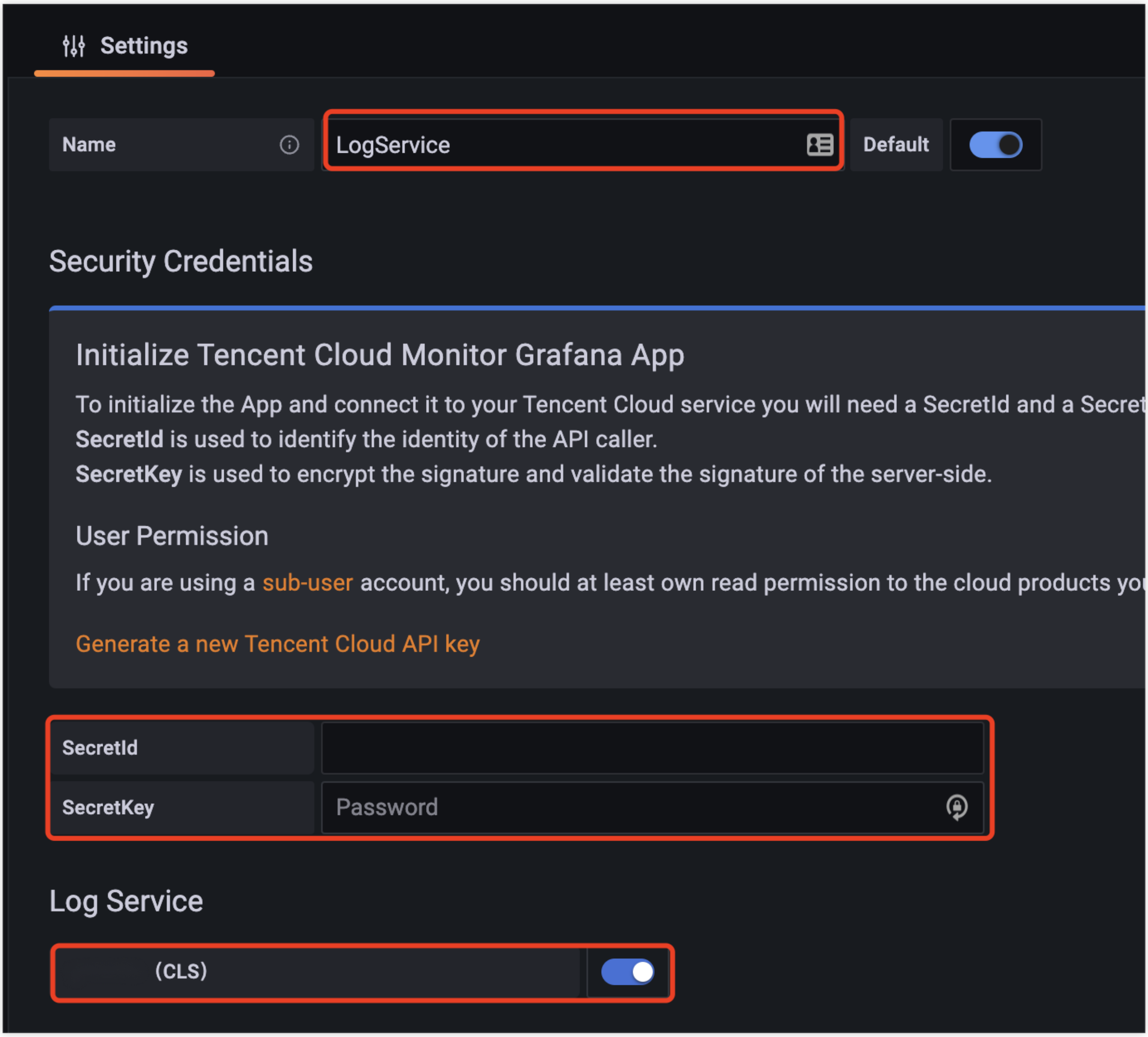
2. Run the following command to restart the Grafana service:

```
service grafana-server restart
```

Configuring log data source

1. Enter `http://${Grafana IP address}:3000` (3000 is the default port number) in your browser to log in to Grafana.
2. On the left sidebar, select the **Settings** icon to go to the **Plugins** page. Select **Tencent Cloud Monitor** and click **Enable** on the **Config** page to enable the plugin.
3. On the **Data Sources** page, click **Add data source**.

4. Select **Tencent Cloud Monitor**, enter the data source name and Tencent Cloud access key as instructed, select **Cloud Log Service (cls)**, and click **Save**.



Settings

Name: LogService

Default

Security Credentials

Initialize Tencent Cloud Monitor Grafana App

To initialize the App and connect it to your Tencent Cloud service you will need a SecretId and a Secret
SecretId is used to identify the identity of the API caller.
SecretKey is used to encrypt the signature and validate the signature of the server-side.

User Permission

If you are using a sub-user account, you should at least own read permission to the cloud products yo

Generate a new Tencent Cloud API key

SecretId

SecretKey: Password

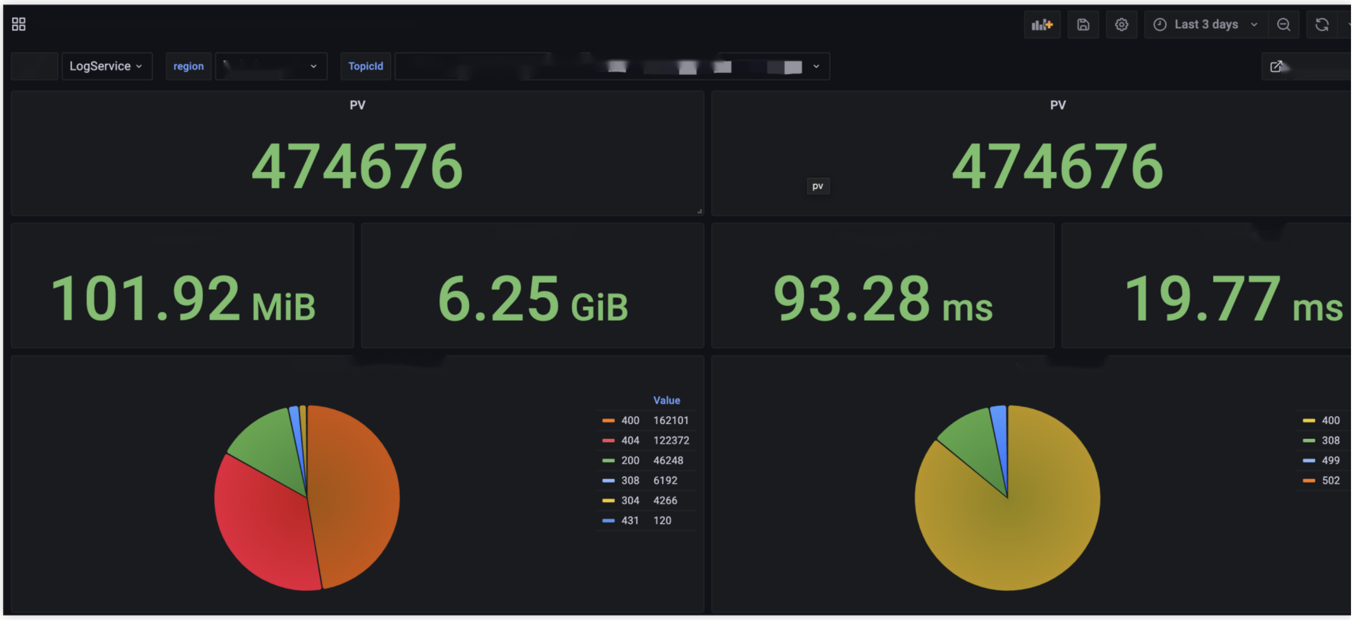
Log Service

(CLS)

Trying out the CLB-DEMO preset dashboard

To quickly try out Tencent Cloud Monitor Grafana App, we recommend you use the demo log feature.

After creating a log topic, you can go to the preset `CLB Demo Access Log` dashboard to view preset content.



Manually configuring dashboard

1. On the left sidebar, click **Create Dashboards**.
2. On the dashboard page, click **Add new panel**.
3. Select a data source, a **region**, and a **log topic**, and enter the corresponding search and analysis statement.
4. Click the time in the upper-right corner and refresh. Then you can view the requested data on the dashboard.

Using log analysis to enter SQL statements for chart drawing and display

For the native charts on Grafana, data visualization is driven by data type. You can determine if and how a chart can be drawn based on the field type.

1. Draw a table.

Tables have no special requirements for data formats, and any content returned for the SQL statement can be displayed.

2. Draw a sequence diagram. The content returned for the SQL statement contains two fields, `analytic_time` of the time type and `log_count` of the numeric type, which can be used to complete the drawing.

```
* | select histogram( cast(__TIMESTAMP__ as timestamp),interval 1 minute) as analyt
```

3. Draw pie charts (Pie, Gauge, BarGauge, or Stat) (note that the `Value Options - show` chart configuration item on the right must be set to `All values`).

```
* | select try_cast(status as varchar) as status, count(*) as log_count group by st
```

The content returned for the SQL statement contains two fields: `status` of the character type and `log_count` of the numeric type. Grafana automatically selects the character type as the tag for drawing.

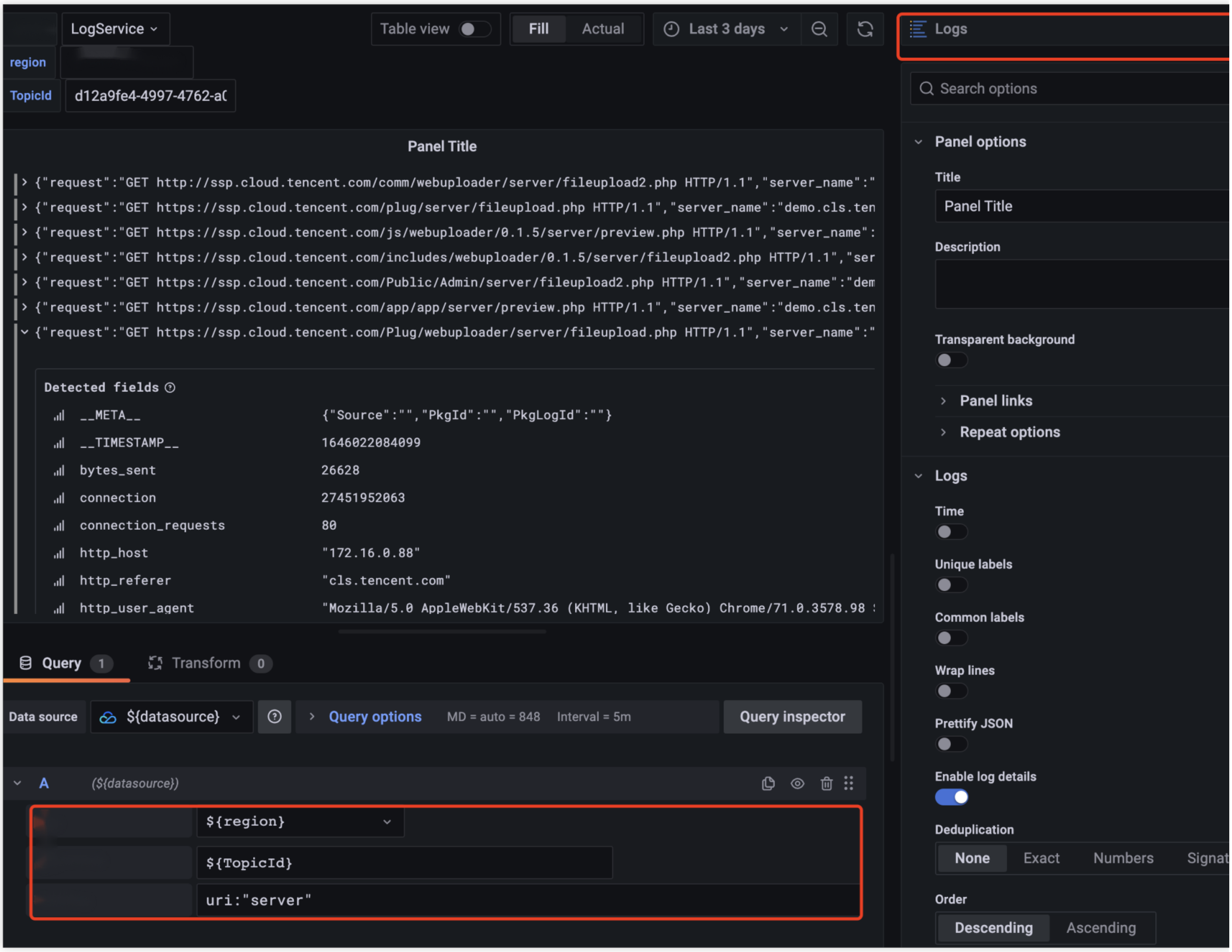
Note:

If `try_cast` is not used for type conversion, the chart drawing effect will be affected.

If the content field types returned for the SQL statement do not meet your chart drawing requirements, in addition to using the type conversion function in SQL, you can use the **Convert field type** feature on the **Transform** tab page of Grafana for visual display.

Viewing raw logs

For search scenarios, we recommend you use the Logs plugin to display data.



region [redacted]

TopicId d12a9fe4-4997-4762-a(

Panel Title

```
> {"request":"GET http://ssp.cloud.tencent.com/comm/webuploader/server/fileupload2.php HTTP/1.1","server_name":"
> {"request":"GET https://ssp.cloud.tencent.com/plug/server/fileupload.php HTTP/1.1","server_name":"demo.cls.ten
> {"request":"GET https://ssp.cloud.tencent.com/js/webuploader/0.1.5/server/preview.php HTTP/1.1","server_name":
> {"request":"GET https://ssp.cloud.tencent.com/includes/webuploader/0.1.5/server/fileupload2.php HTTP/1.1","ser
> {"request":"GET https://ssp.cloud.tencent.com/Public/Admin/server/fileupload2.php HTTP/1.1","server_name":"dem
> {"request":"GET https://ssp.cloud.tencent.com/app/app/server/preview.php HTTP/1.1","server_name":"demo.cls.ten
v {"request":"GET https://ssp.cloud.tencent.com/Plug/webuploader/server/fileupload.php HTTP/1.1","server_name":"

  Detected fields
    __META__                       {"Source":"","PkgId":"","PkgLogId":""}
    __TIMESTAMP__                  1646022084099
    bytes_sent                     26628
    connection                     27451952063
    connection_requests            80
    http_host                      "172.16.0.88"
    http_referer                   "cls.tencent.com"
    http_user_agent                "Mozilla/5.0 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 :
```

Query 1 Transform 0

Data source ${datasource} Query options MD = auto = 848 Interval = 5m Query inspector

A (${datasource})

[redacted] ${region}

[redacted] ${TopicId}

[redacted] uri:"server"

Panel options

Title: Panel Title

Description

Transparent background ○

Panel links

Repeat options

Logs

Time ○

Unique labels ○

Common labels ○

Wrap lines ○

Prettify JSON ○

Enable log details ●

Deduplication: [✓ None] Exact Numbers Signat

Order: [✓ Descending] Ascending

If you are using Grafana v8.3 or later, you can also use the **Extract fields** feature on the **Transform** tab page of Grafana to extract fields from the content returned for the search and view data in **Table**.

Other Usage Guidelines for the Plugin

Git repository for Tencent Cloud Monitor Grafana App

CLS documentation

CM data source template variables