

How do I use CloudTrail to track API calls to my Amazon EC2 instances?

By AWS Official

Published: 2019-09-04 · Archived: 2026-04-06 03:33:13 UTC

I want to track API calls that run, stop, start, and terminate my Amazon Elastic Compute Cloud (Amazon EC2) instances.

Resolution

You can use AWS CloudTrail to track API calls to your AWS account. These API calls, also known as event types, include [RunInstances](#), [StopInstances](#), [StartInstances](#), and [TerminateInstances](#).

Note: You can use CloudTrail to search event history for the last 90 days.

Track API calls with CloudTrail event history

1. Open the [CloudTrail console](#).
2. Choose **Event history**.
3. For **Filter**, select the dropdown list. Then, choose **Event name**.
4. For **Enter event name**, enter the event type. Then, choose the event type.
5. For **Time range**, specify the search period.
6. Choose **Apply**.

For more information, see [Working with CloudTrail event history](#) and [Viewing recent management events with the console](#).

Track API calls with Athena queries

You can use the following example queries for the [RunInstances](#) API call and adapt them for other supported event types. For more information, see [How do I automatically create tables in Athena to search through CloudTrail logs?](#)

Important: Replace **cloudtrail-logs** with your Amazon Athena table name in the following example queries.

Example query to return all available event information for the RunInstances API call:

```
SELECT *  
FROM cloudtrail-logs
```

```
WHERE eventName = 'RunInstances'
```

Example query to return filtered event information for the RunInstances API call:

```
SELECT userIdentity.username, eventTime, eventName
FROM cloudtrail-logs
WHERE eventName = 'RunInstances'
```

Example query to find event information for API calls that end with "Instances" during a specific time period:

Important: Replace **2021-07-01T00:00:01Z** with the start date.

```
SELECT userIdentity.username, eventTime, eventName
FROM cloudtrail-logs
WHERE (eventName LIKE '%Instances') AND eventTime > '2021-07-01T00:00:01Z'
```

Track API calls with archived CloudWatch logs in Amazon S3

Prerequisite: To log events to an Amazon Simple Storage Service (Amazon S3) bucket, you must [create a CloudTrail trail](#).

1. To access your CloudTrail log files, follow the instructions in [Getting and viewing your CloudTrail log files](#).
2. Download your log files. For instructions, see [Downloading your CloudTrail log files](#).
3. Search the logs with jq or another JSON command line processor to find event types.

Example jq procedure to search Amazon CloudWatch logs from Amazon S3 for event types:

Open a Bash terminal, and then run the following command to create a storage directory.

```
$ mkdir cloudtrail-logs
```

Navigate to the new directory. Then, run the following command to download the CloudTrail logs:

Important: Replace the example **my_cloudtrail_bucket** with your S3 bucket.

```
$ cd cloudtrail-logs  
$ aws s3 cp s3://my_cloudtrail_bucket/AWSLogs/012345678901/CloudTrail/eu-west-1/2019/08/07 ./ --recursive
```

Run the following `gzip` command to decompress the log files:

Important: Replace `*` with the file name that you want to decompress.

```
$ gzip -d *
```

Example `jq` query to return all available event information for the `RunInstances` API call:

Run the following `jq` query to find specific event types.

```
cat * | jq '.Records[] | select(.eventName=="RunInstances")'
```

Example `jq` query to return all available event information for the `StopInstances` and `TerminateInstances` API calls:

```
cat * | jq '.Records[] | select(.eventName=="StopInstances" or .eventName=="TerminateInstances" )'
```

Related information

[How do I use CloudTrail to review what API calls and actions have occurred in my AWS account?](#)

[Creating metrics from log events using filters](#)

[AWS Config console now displays API events associated with configuration changes](#)

Source: <https://aws.amazon.com/premiumsupport/knowledge-center/cloudtrail-search-api-calls/>