



API Reference

AWS CloudTrail



API Version 2021-08-11

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS CloudTrail: API Reference

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
Actions	2
PutAuditEvents	3
Request Syntax	3
URI Request Parameters	3
Request Body	4
Response Syntax	4
Response Elements	4
Errors	5
See Also	6
Data Types	7
AuditEvent	8
Contents	8
See Also	8
AuditEventResultEntry	10
Contents	10
See Also	10
ResultErrorEntry	11
Contents	11
See Also	11
Common Parameters	13
Common Error Types	16

Welcome

The CloudTrail Data Service lets you ingest events into CloudTrail from any source in your hybrid environments, such as in-house or SaaS applications hosted on-premises or in the cloud, virtual machines, or containers. You can store, access, analyze, troubleshoot and take action on this data without maintaining multiple log aggregators and reporting tools. After you run `PutAuditEvents` to ingest your application activity into CloudTrail, you can use CloudTrail Lake to search, query, and analyze the data that is logged from your applications.

This document was last published on September 18, 2026.

Actions

The following actions are supported:

- [PutAuditEvents](#)

PutAuditEvents

Ingests your application events into CloudTrail Lake. A required parameter, `auditEvents`, accepts the JSON records (also called *payload*) of events that you want CloudTrail to ingest. You can add up to 100 of these events (or up to 1 MB) per `PutAuditEvents` request.

Request Syntax

```
POST /PutAuditEvents?channelArn=channelArn&externalId=externalId HTTP/1.1
Content-type: application/json
```

```
{
  "auditEvents": [
    {
      "eventData": "string",
      "eventDataChecksum": "string",
      "id": "string"
    }
  ]
}
```

URI Request Parameters

The request uses the following URI parameters.

channelArn

The ARN or ID (the ARN suffix) of a channel.

Pattern: `arn:.*`

Required: Yes

externalId

A unique identifier that is conditionally required when the channel's resource policy includes an external ID. This value can be any string, such as a passphrase or account number.

Length Constraints: Minimum length of 2. Maximum length of 1224.

Pattern: `[\w+=, .@:\-]*`

Request Body

The request accepts the following data in JSON format.

auditEvents

The JSON payload of events that you want to ingest. You can also point to the JSON event payload in a file.

Type: Array of [AuditEvent](#) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "failed": [
    {
      "errorCode": "string",
      "errorMessage": "string",
      "id": "string"
    }
  ],
  "successful": [
    {
      "eventID": "string",
      "id": "string"
    }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

failed

Lists events in the provided event payload that could not be ingested into CloudTrail, and includes the error code and error message returned for events that could not be ingested.

Type: Array of [ResultErrorEntry](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

successful

Lists events in the provided event payload that were successfully ingested into CloudTrail.

Type: Array of [AuditEventResultEntry](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Errors

For information about the errors that are common to all actions, see [Common Error Types](#).

ChannelInsufficientPermission

The caller's account ID must be the same as the channel owner's account ID.

HTTP Status Code: 400

ChannelNotFound

The channel could not be found.

HTTP Status Code: 400

ChannelUnsupportedSchema

The schema type of the event is not supported.

HTTP Status Code: 400

DuplicatedAuditEventId

Two or more entries in the request have the same event ID.

HTTP Status Code: 400

InvalidChannelARN

The specified channel ARN is not a valid channel ARN.

HTTP Status Code: 400

UnsupportedOperationException

The operation requested is not supported in this region or account.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python \(Boto3\)](#)
- [AWS SDK for Ruby V3](#)

Data Types

The AWS CloudTrail Data Service API contains several data types that various actions use. This section describes each data type in detail.

Note

The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [AuditEvent](#)
- [AuditEventResultEntry](#)
- [ResultErrorEntry](#)

AuditEvent

An event from a source outside of AWS that you want CloudTrail to log.

Contents

eventData

The content of an audit event that comes from the event, such as `userIdentity`, `userAgent`, and `eventSource`.

Type: String

Required: Yes

id

The original event ID from the source event.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[-_A-Za-z0-9]+`

Required: Yes

eventDataChecksum

A checksum is a base64-SHA256 algorithm that helps you verify that CloudTrail receives the event that matches with the checksum. Calculate the checksum by running a command like the following:

```
printf %s $eventdata | openssl dgst -binary -sha256 | base64
```

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

AuditEventResultEntry

A response that includes successful and failed event results.

Contents

eventID

The event ID assigned by CloudTrail.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[-_A-Za-z0-9]+`

Required: Yes

id

The original event ID from the source event.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[-_A-Za-z0-9]+`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResultErrorEntry

Includes the error code and error message for events that could not be ingested by CloudTrail.

Contents

errorCode

The error code for events that could not be ingested by CloudTrail. Possible error codes include: `FieldTooLong`, `FieldNotFound`, `InvalidChecksum`, `InvalidData`, `InvalidRecipient`, `InvalidEventSource`, `AccountNotSubscribed`, `Throttling`, and `InternalFailure`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

errorMessage

The message that describes the error for events that could not be ingested by CloudTrail.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Required: Yes

id

The original event ID from the source event that could not be ingested by CloudTrail.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[-_A-Za-z0-9]+`

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see [Signing AWS API requests](#) in the *IAM User Guide*.

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key/YYYYMMDD/region/service/aws4_request*.

For more information, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Elements of an AWS API request signature](#) in the *IAM User Guide*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see [AWS services that work with IAM](#) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Error Types

This section lists common error types that this AWS service may return. Not all services return all error types listed here. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You don't have permission to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 403

ExpiredTokenException

The security token included in the request has expired. Request a new security token and try again.

HTTP Status Code: 403

IncompleteSignature

The request signature doesn't conform to AWS standards. Verify that you're using valid AWS credentials and that your request is properly formatted. If you're using an SDK, ensure it's up to date.

HTTP Status Code: 403

InternalFailure

The request can't be processed right now because of an internal server issue. Try again later. If the problem persists, contact AWS Support.

HTTP Status Code: 500

MalformedHttpRequestException

The request body can't be processed. This typically happens when the request body can't be decompressed using the specified content encoding algorithm. Verify that the content encoding header matches the compression format used.

HTTP Status Code: 400

NotAuthorized

You don't have permissions to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 401

OptInRequired

Your AWS account needs a subscription for this service. Verify that you've enabled the service in your account.

HTTP Status Code: 403

RequestAbortedException

The request was aborted before a response could be returned. This typically happens when the client closes the connection.

HTTP Status Code: 400

RequestEntityTooLargeException

The request entity is too large. Reduce the size of the request body and try again.

HTTP Status Code: 413

RequestTimeoutException

The request timed out. The server didn't receive the complete request within the expected time frame. Try again.

HTTP Status Code: 408

ServiceUnavailable

The service is temporarily unavailable. Try again later.

HTTP Status Code: 503

ThrottlingException

Your request rate is too high. The AWS SDKs automatically retry requests that receive this exception. Reduce the frequency of requests.

HTTP Status Code: 400

UnknownOperationException

The action or operation isn't recognized. Verify that the action name is spelled correctly and that it's supported by the API version you're using.

HTTP Status Code: 404

UnrecognizedClientException

The X.509 certificate or AWS access key ID you provided doesn't exist in our records. Verify that you're using valid credentials and that they haven't expired.

HTTP Status Code: 403

ValidationError

The input doesn't meet the required format or constraints. Check that all required parameters are included and that values are valid.

HTTP Status Code: 400