



API Reference

# AWS CloudTrail



**API Version 2021-08-11**

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS CloudTrail: API Reference

Copyright © 2025 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.

This documentation is a draft for private preview for regions in the AWS European Sovereign Cloud. Documentation content will continue to evolve. Published: January 1, 2026.

---

# Table of Contents

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

# 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 January 1, 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 Errors](#).

### **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](#)
- [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](#)
- [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*.

## Action

The action to be performed.

Type: string

Required: Yes

## Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

## 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 Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

## **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 403

## **ExpiredTokenException**

The security token included in the request is expired

HTTP Status Code: 403

## **IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 403

## **InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

## **MalformedHttpRequestException**

Problems with the request at the HTTP level, e.g. we can't decompress the body according to the decompression algorithm specified by the content-encoding.

HTTP Status Code: 400

## **NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 401

## **OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

### **RequestAbortedException**

Convenient exception that can be used when a request is aborted before a reply is sent back (e.g. client closed connection).

HTTP Status Code: 400

### **RequestEntityTooLargeException**

Problems with the request at the HTTP level. The request entity is too large.

HTTP Status Code: 413

### **RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

### **RequestTimeoutException**

Problems with the request at the HTTP level. Reading the Request timed out.

HTTP Status Code: 408

### **ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

### **UnrecognizedClientException**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

## **UnknownOperationException**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 404

## **ValidationError**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400