AWS X-Ray: API Reference
Copyright © 2018 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  
BatchGetTraces .................................................................................................................................. 3  
  Request Syntax ................................................................................................................................. 3  
  URI Request Parameters ................................................................................................................... 3  
  Request Body ................................................................................................................................... 3  
  Response Syntax ............................................................................................................................. 3  
  Response Elements .......................................................................................................................... 4  
  Errors ............................................................................................................................................ 4  
  See Also ......................................................................................................................................... 4  
GetServiceGraph .............................................................................................................................. 6  
  Request Syntax ................................................................................................................................. 6  
  URI Request Parameters ................................................................................................................... 6  
  Request Body ................................................................................................................................... 6  
  Response Syntax ............................................................................................................................. 6  
  Response Elements .......................................................................................................................... 8  
  Errors ............................................................................................................................................ 8  
  See Also ......................................................................................................................................... 8  
GetTraceGraph .................................................................................................................................... 10  
  Request Syntax ................................................................................................................................. 10  
  URI Request Parameters .................................................................................................................. 10  
  Request Body .................................................................................................................................. 10  
  Response Syntax ............................................................................................................................. 10  
  Response Elements .......................................................................................................................... 12  
  Errors ............................................................................................................................................ 12  
  See Also ......................................................................................................................................... 12  
GetTraceSummaries ......................................................................................................................... 13  
  Request Syntax ................................................................................................................................. 13  
  URI Request Parameters .................................................................................................................. 13  
  Request Body .................................................................................................................................. 13  
  Response Syntax ............................................................................................................................. 14  
  Response Elements .......................................................................................................................... 15  
  Errors ............................................................................................................................................ 16  
  See Also ......................................................................................................................................... 16  
PutTelemetryRecords ....................................................................................................................... 17  
  Request Syntax ................................................................................................................................. 17  
  URI Request Parameters .................................................................................................................. 17  
  Request Body .................................................................................................................................. 17  
  Response Syntax ............................................................................................................................. 18  
  Response Elements .......................................................................................................................... 18  
  Errors ............................................................................................................................................ 18  
  See Also ......................................................................................................................................... 18  
PutTraceSegments ............................................................................................................................ 19  
  Request Syntax ................................................................................................................................. 19  
  URI Request Parameters .................................................................................................................. 19  
  Request Body .................................................................................................................................. 19  
  Response Syntax ............................................................................................................................. 20  
  Response Elements .......................................................................................................................... 20  
  Errors ............................................................................................................................................ 20  
  See Also ......................................................................................................................................... 20  
Data Types ........................................................................................................................................... 22  
  Alias .............................................................................................................................................. 23  
  Contents ......................................................................................................................................... 23  
  See Also ......................................................................................................................................... 23
Welcome

AWS X-Ray provides APIs for managing debug traces and retrieving service maps and other data created by processing those traces.

This document was last published on February 7, 2018.
Actions

The following actions are supported:

- BatchGetTraces (p. 3)
- GetServiceGraph (p. 6)
- GetTraceGraph (p. 10)
- GetTraceSummaries (p. 13)
- PutTelemetryRecords (p. 17)
- PutTraceSegments (p. 19)
BatchGetTraces

Retrieves a list of traces specified by ID. Each trace is a collection of segment documents that originates from a single request. Use GetTraceSummaries to get a list of trace IDs.

Request Syntax

```plaintext
POST /Traces HTTP/1.1
Content-type: application/json

{
   "NextToken": "string",
   "TraceIds": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

**NextToken (p. 3)**

- Pagination token. Not used.
- Type: String
- Required: No

**TraceIds (p. 3)**

- Specify the trace IDs of requests for which to retrieve segments.
- Type: Array of strings
- Required: Yes

Response Syntax

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

{
   "NextToken": "string",
   "Traces": [
      {
         "Duration": number,
         "Id": "string",
         "Segments": [
            {
               "Document": "string",
               "Id": "string"
            }
      }

API Version 2016-04-12
3
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.

**NextToken (p. 3)**

Pagination token. Not used.

Type: String

**Traces (p. 3)**

Full traces for the specified requests.

Type: Array of Trace (p. 40) objects

**UnprocessedTraceIds (p. 3)**

Trace IDs of requests that haven’t been processed.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 35.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

**InvalidRequestException**

The request is missing required parameters or has invalid parameters.

HTTP Status Code: 400

**ThrottledException**

The request exceeds the maximum number of requests per second.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
See Also

- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
GetServiceGraph

Retrieves a document that describes services that process incoming requests, and downstream services that they call as a result. Root services process incoming requests and make calls to downstream services. Root services are applications that use the AWS X-Ray SDK. Downstream services can be other applications, AWS resources, HTTP web APIs, or SQL databases.

Request Syntax

```
POST /ServiceGraph HTTP/1.1
Content-type: application/json

{
  "EndTime": number,
  "NextToken": "string",
  "StartTime": number
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

EndTime (p. 6)

The end of the time frame for which to generate a graph.
Type: Timestamp
Required: Yes

NextToken (p. 6)

Pagination token. Not used.
Type: String
Required: No

StartTime (p. 6)

The start of the time frame for which to generate a graph.
Type: Timestamp
Required: Yes

Response Syntax

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

{
  "EndTime": number,
}
```
"NextToken": "string",
"Services": [
  {
    "AccountId": "string",
    "DurationHistogram": [
      {
        "Count": number,
        "Value": number
      }
    ],
    "Edges": [
      {
        "Aliases": [
          {
            "Name": "string",
            "Names": [ "string" ],
            "Type": "string"
          }
        ],
        "EndTime": number,
        "ReferenceId": number,
        "ResponseTimeHistogram": [
          {
            "Count": number,
            "Value": number
          }
        ],
        "StartTime": number,
        "SummaryStatistics": {
          "ErrorStatistics": {
            "OtherCount": number,
            "ThrottleCount": number,
            "TotalCount": number
          },
          "FaultStatistics": {
            "OtherCount": number,
            "TotalCount": number
          },
          "OkCount": number,
          "TotalCount": number,
          "TotalResponseTime": number
        }
      }
    ],
    "EndTime": number,
    "Name": "string",
    "Names": [ "string" ],
    "ReferenceId": number,
    "ResponseTimeHistogram": [
      {
        "Count": number,
        "Value": number
      }
    ],
    "Root": boolean,
    "StartTime": number,
    "State": "string",
    "SummaryStatistics": {
      "ErrorStatistics": {
        "OtherCount": number,
        "ThrottleCount": number,
        "TotalCount": number
      },
      "FaultStatistics": {
        "OtherCount": number,
        "TotalCount": number
      }
    }
  }
],
"EndTime": number,
"Name": "string",
"Names": [ "string" ],
"ReferenceId": number,
"ResponseTimeHistogram": [
  {
    "Count": number,
    "Value": number
  }
],
"Root": boolean,
"StartTime": number,
"State": "string",
"SummaryStatistics": {
  "ErrorStatistics": {
    "OtherCount": number,
    "ThrottleCount": number,
    "TotalCount": number
  },
  "FaultStatistics": {
    "OtherCount": number,
    "TotalCount": number
  }
}
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.

EndTime (p. 6)

   The end of the time frame for which the graph was generated.
   Type: Timestamp

NextToken (p. 6)

   Pagination token. Not used.
   Type: String

Services (p. 6)

   The services that have processed a traced request during the specified time frame.
   Type: Array of Service (p. 34) objects

StartTime (p. 6)

   The start of the time frame for which the graph was generated.
   Type: Timestamp

Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

InvalidRequestException

   The request is missing required parameters or has invalid parameters.

   HTTP Status Code: 400

ThrottledException

   The request exceeds the maximum number of requests per second.

   HTTP Status Code: 429

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
GetTraceGraph

Retrieves a service graph for one or more specific trace IDs.

Request Syntax

```plaintext
POST /TraceGraph HTTP/1.1
Content-type: application/json

{
  "NextToken": "string",
  "TraceIds": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

- **NextToken (p. 10)**
  - Pagination token. Not used.
  - Type: String
  - Required: No

- **TraceIds (p. 10)**
  - Trace IDs of requests for which to generate a service graph.
  - Type: Array of strings
  - Required: Yes

Response Syntax

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

{
  "NextToken": "string",
  "Services": [
    {
      "AccountId": "string",
      "DurationHistogram": [
        {
          "Count": number,
          "Value": number
```


```
},
  "Edges": [
    {
      "Aliases": [
        {
          "Name": "string",
          "Names": [ "string" ],
          "Type": "string"
        }
      ],
      "EndTime": number,
      "ReferenceId": number,
      "ResponseTimeHistogram": [
        {
          "Count": number,
          "Value": number
        }
      ],
      "StartTime": number,
      "SummaryStatistics": {
        "ErrorStatistics": {
          "OtherCount": number,
          "ThrottleCount": number,
          "TotalCount": number
        },
        "FaultStatistics": {
          "OtherCount": number,
          "TotalCount": number
        },
        "OkCount": number,
        "TotalCount": number,
        "TotalResponseTime": number
      }
    }
  ],
  "EndTime": number,
  "Name": "string",
  "Names": [ "string" ],
  "ReferenceId": number,
  "ResponseTimeHistogram": [
    {
      "Count": number,
      "Value": number
    }
  ],
  "Root": boolean,
  "StartTime": number,
  "State": "string",
  "SummaryStatistics": {
    "ErrorStatistics": {
      "OtherCount": number,
      "ThrottleCount": number,
      "TotalCount": number
    },
    "FaultStatistics": {
      "OtherCount": number,
      "TotalCount": number
    },
    "OkCount": number,
    "TotalCount": number,
    "TotalResponseTime": number
  },
  "Type": "string"
}
```

API Version 2016-04-12
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.

**NextToken (p. 10)**

Pagination token. Not used.

Type: String

**Services (p. 10)**

The services that have processed one of the specified requests.

Type: Array of Service (p. 34) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

**InvalidRequestException**

The request is missing required parameters or has invalid parameters.

HTTP Status Code: 400

**ThrottledException**

The request exceeds the maximum number of requests per second.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
GetTraceSummaries

Retrieves IDs and metadata for traces available for a specified time frame using an optional filter. To get the full traces, pass the trace IDs to BatchGetTraces.

A filter expression can target traced requests that hit specific service nodes or edges, have errors, or come from a known user. For example, the following filter expression targets traces that pass through api.example.com:

service("api.example.com")

This filter expression finds traces that have an annotation named account with the value 12345:

annotation.account = "12345"

For a full list of indexed fields and keywords that you can use in filter expressions, see Using Filter Expressions in the AWS X-Ray Developer Guide.

Request Syntax

```
POST /TraceSummaries HTTP/1.1
Content-type: application/json
{
   "EndTime": number,
   "FilterExpression": "string",
   "NextToken": "string",
   "Sampling": boolean,
   "StartTime": number
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

**EndTime (p. 13)**

The end of the time frame for which to retrieve traces.

Type: Timestamp

Required: Yes

**FilterExpression (p. 13)**

Specify a filter expression to retrieve trace summaries for services or requests that meet certain requirements.

Type: String


Required: No
NextToken (p. 13)

Specify the pagination token returned by a previous request to retrieve the next page of results.

Type: String

Required: No

Sampling (p. 13)

Set to true to get summaries for only a subset of available traces.

Type: Boolean

Required: No

StartTime (p. 13)

The start of the time frame for which to retrieve traces.

Type: Timestamp

Required: Yes

Response Syntax

HTTP/1.1 200
Content-type: application/json

{
  "ApproximateTime": number,
  "NextToken": "string",
  "TracesProcessedCount": number,
  "TraceSummaries": [
    {
      "Annotations": {
        "string" : [
          {
            "AnnotationValue": {
              "BooleanValue": boolean,
              "NumberValue": number,
              "StringValue": "string"
            },
            "ServiceIds": [
              {
                "AccountId": "string",
                "Name": "string",
                "Names": [ "string" ],
                "Type": "string"
              }
            ]
          }
        }
      },
      "Duration": number,
      "HasError": boolean,
      "HasFault": boolean,
      "HasThrottle": boolean,
      "Http": {
        "ClientIp": "string",
        "HttpMethod": "string",
        "HttpStatus": number,
      }
    }
  ]
}
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.

**ApproximateTime (p. 14)**

The start time of this page of results.

Type: Timestamp

**NextToken (p. 14)**

If the requested time frame contained more than one page of results, you can use this token to retrieve the next page. The first page contains the most most recent results, closest to the end of the time frame.

Type: String

**TracesProcessedCount (p. 14)**

The total number of traces processed, including traces that did not match the specified filter expression.

Type: Long

**TraceSummaries (p. 14)**

Trace IDs and metadata for traces that were found in the specified time frame.

Type: Array of TraceSummary (p. 41) objects
Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

InvalidRequestException

The request is missing required parameters or has invalid parameters.

HTTP Status Code: 400

ThrottledException

The request exceeds the maximum number of requests per second.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
PutTelemetryRecords

Used by the AWS X-Ray daemon to upload telemetry.

Request Syntax

```
POST /TelemetryRecords HTTP/1.1
Content-type: application/json

{
    "EC2InstanceId": "string",
    "Hostname": "string",
    "ResourceARN": "string",
    "TelemetryRecords": [
        {
            "BackendConnectionErrors": {
                "ConnectionRefusedCount": number,
                "HTTPCode4XXCount": number,
                "HTTPCode5XXCount": number,
                "OtherCount": number,
                "UnknownHostCount": number
            },
            "SegmentsReceivedCount": number,
            "SegmentsRejectedCount": number,
            "SegmentsSentCount": number,
            "SegmentsSpilloverCount": number,
            "Timestamp": number
        }
    ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

EC2InstanceId (p. 17)

Type: String
Length Constraints: Maximum length of 20.
Required: No

Hostname (p. 17)

Type: String
Length Constraints: Maximum length of 255.
Required: No

ResourceARN (p. 17)
Type: String
Length Constraints: Maximum length of 500.
Required: No
TelemetryRecords (p. 17)
Type: Array of TelemetryRecord (p. 39) objects
Required: Yes

Response Syntax

HTTP/1.1 200

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

InvalidRequestException

The request is missing required parameters or has invalid parameters.

HTTP Status Code: 400

ThrottledException

The request exceeds the maximum number of requests per second.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
PutTraceSegments

Upads segment documents to AWS X-Ray. The X-Ray SDK generates segment documents and sends them to the X-Ray daemon, which uploads them in batches. A segment document can be a completed segment, an in-progress segment, or an array of subsegments.

Segments must include the following fields. For the full segment document schema, see AWS X-Ray Segment Documents in the AWS X-Ray Developer Guide.

**Required Segment Document Fields**

- **name** - The name of the service that handled the request.
- **id** - A 64-bit identifier for the segment, unique among segments in the same trace, in 16 hexadecimal digits.
- **trace_id** - A unique identifier that connects all segments and subsegments originating from a single client request.
- **start_time** - Time the segment or subsegment was created, in floating point seconds in epoch time, accurate to milliseconds. For example, 1480615200.010 or 1.4806152000010E9.
- **end_time** - Time the segment or subsegment was closed. For example, 1480615200.090 or 1.480615200090E9. Specify either an end_time or in_progress.
- **in_progress** - Set to true instead of specifying an end_time to record that a segment has been started, but is not complete. Send an in progress segment when your application receives a request that will take a long time to serve, to trace the fact that the request was received. When the response is sent, send the complete segment to overwrite the in-progress segment.

A trace_id consists of three numbers separated by hyphens. For example, 1-58406520-a006649127e371903a2de979. This includes:

**Trace ID Format**

- The version number, i.e. 1.
- The time of the original request, in Unix epoch time, in 8 hexadecimal digits. For example, 10:00AM December 2nd, 2016 PST in epoch time is 1480615200 seconds, or 58406520 in hexadecimal.
- A 96-bit identifier for the trace, globally unique, in 24 hexadecimal digits.

**Request Syntax**

```
POST /TraceSegments HTTP/1.1
Content-type: application/json

{  
  "TraceSegmentDocuments": [ "string" ]
}
```

**URI Request Parameters**

The request does not use any URI parameters.

**Request Body**

The request accepts the following data in JSON format.
TraceSegmentDocuments (p. 19)

A string containing a JSON document defining one or more segments or subsegments.

Type: Array of strings

Required: Yes

Response Syntax

HTTP/1.1 200
Content-type: application/json

{  
  "UnprocessedTraceSegments": [
  {  
    "ErrorCode": "string",
    "Id": "string",
    "Message": "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.

UnprocessedTraceSegments (p. 20)

Segments that failed processing.

Type: Array of UnprocessedTraceSegment (p. 44) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 48).

InvalidRequestException

The request is missing required parameters or has invalid parameters.

HTTP Status Code: 400

ThrottledException

The request exceeds the maximum number of requests per second.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
Data Types

The AWS X-Ray 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:

- Alias (p. 23)
- AnnotationValue (p. 24)
- BackendConnectionErrors (p. 25)
- Edge (p. 26)
- EdgeStatistics (p. 28)
- ErrorStatistics (p. 29)
- FaultStatistics (p. 30)
- HistogramEntry (p. 31)
- Http (p. 32)
- Segment (p. 33)
- Service (p. 34)
- ServiceId (p. 37)
- ServiceStatistics (p. 38)
- TelemetryRecord (p. 39)
- Trace (p. 40)
- TraceSummary (p. 41)
- TraceUser (p. 43)
- UnprocessedTraceSegment (p. 44)
- ValueWithServiceIds (p. 45)
Alias

An alias for an edge.

Contents

Name

The canonical name of the alias.

Type: String

Required: No

Names

A list of names for the alias, including the canonical name.

Type: Array of strings

Required: No

Type

The type of the alias.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
AnnotationValue

Value of a segment annotation. Has one of three value types: Number, Boolean or String.

Contents

BooleanValue

Value for a Boolean annotation.

Type: Boolean
Required: No

NumberValue

Value for a Number annotation.

Type: Double
Required: No

StringValue

Value for a String annotation.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
BackendConnectionErrors

Contents

ConnectionRefusedCount

Type: Integer
Required: No

HTTPCode4XXCount

Type: Integer
Required: No

HTTPCode5XXCount

Type: Integer
Required: No

OtherCount

Type: Integer
Required: No

TimeoutCount

Type: Integer
Required: No

UnknownHostCount

Type: Integer
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Edge

Information about a connection between two services.

Contents

Aliases

Aliases for the edge.

Type: Array of Alias (p. 23) objects

Required: No

EndTime

The end time of the last segment on the edge.

Type: Timestamp

Required: No

ReferenceId

Identifier of the edge. Unique within a service map.

Type: Integer

Required: No

ResponseTimeHistogram

A histogram that maps the spread of client response times on an edge.

Type: Array of HistogramEntry (p. 31) objects

Required: No

StartTime

The start time of the first segment on the edge.

Type: Timestamp

Required: No

SummaryStatistics

Response statistics for segments on the edge.

Type: EdgeStatistics (p. 28) object

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 Go
- AWS SDK for Java
• AWS SDK for Ruby V2
EdgeStatistics

Response statistics for an edge.

Contents

ErrorStatistics

Information about requests that failed with a 4xx Client Error status code.

Type: ErrorStatistics (p. 29) object

Required: No

FaultStatistics

Information about requests that failed with a 5xx Server Error status code.

Type: FaultStatistics (p. 30) object

Required: No

OkCount

The number of requests that completed with a 2xx Success status code.

Type: Long

Required: No

TotalCount

The total number of completed requests.

Type: Long

Required: No

TotalResponseTime

The aggregate response time of completed requests.

Type: Double

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
ErrorStatistics

Information about requests that failed with a 4xx Client Error status code.

Contents

OtherCount

The number of requests that failed with untracked 4xx Client Error status codes.

Type: Long
Required: No

ThrottleCount

The number of requests that failed with a 419 throttling status code.

Type: Long
Required: No

TotalCount

The total number of requests that failed with a 4xx Client Error status code.

Type: Long
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
FaultStatistics

Information about requests that failed with a 5xx Server Error status code.

Contents

OtherCount

The number of requests that failed with untracked 5xx Server Error status codes.

Type: Long
Required: No

TotalCount

The total number of requests that failed with a 5xx Server Error status code.

Type: Long
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
HistogramEntry

An entry in a histogram for a statistic. A histogram maps the range of observed values on the X axis, and the prevalence of each value on the Y axis.

Contents

Count

The prevalence of the entry.

Type: Integer

Required: No

Value

The value of the entry.

Type: Double

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Http

Information about an HTTP request.

Contents

ClientIp
  The IP address of the requestor.
  Type: String
  Required: No

HttpMethod
  The request method.
  Type: String
  Required: No

HttpStatus
  The response status.
  Type: Integer
  Required: No

HttpURL
  The request URL.
  Type: String
  Required: No

UserAgent
  The request’s user agent string.
  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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Segment

A segment from a trace that has been ingested by the X-Ray service. The segment can be compiled from documents uploaded with PutTraceSegments (p. 19), or an inferred segment for a downstream service, generated from a subsegment sent by the service that called it.

For the full segment document schema, see AWS X-Ray Segment Documents in the AWS X-Ray Developer Guide.

Contents

Document

The segment document.
Type: String
Length Constraints: Minimum length of 1.
Required: No

Id

The segment's ID.
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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Service

Information about an application that processed requests, users that made requests, or downstream services, resources and applications that an application used.

Contents

AccountId

Identifier of the AWS account in which the service runs.
Type: String
Required: No

DurationHistogram

A histogram that maps the spread of service durations.
Type: Array of HistogramEntry (p. 31) objects
Required: No

Edges

Connections to downstream services.
Type: Array of Edge (p. 26) objects
Required: No

EndTime

The end time of the last segment that the service generated.
Type: Timestamp
Required: No

Name

The canonical name of the service.
Type: String
Required: No

Names

A list of names for the service, including the canonical name.
Type: Array of strings
Required: No

ReferenceId

Identifier for the service. Unique within the service map.
Type: Integer
Required: No
ResponseTimeHistogram

A histogram that maps the spread of service response times.

Type: Array of HistogramEntry (p. 31) objects

Required: No

Root

Indicates that the service was the first service to process a request.

Type: Boolean

Required: No

StartTime

The start time of the first segment that the service generated.

Type: Timestamp

Required: No

State

The service's state.

Type: String

Required: No

SummaryStatistics

Aggregated statistics for the service.

Type: ServiceStatistics (p. 38) object

Required: No

Type

The type of service.

- AWS Resource - The type of an AWS resource. For example, `AWS::EC2::Instance` for an application running on Amazon EC2 or `AWS::DynamoDB::Table` for an Amazon DynamoDB table that the application used.
- AWS Service - The type of an AWS service. For example, `AWS::DynamoDB` for downstream calls to Amazon DynamoDB that didn't target a specific table.
- client - Represents the clients that sent requests to a root service.
- remote - A downstream service of indeterminate type.

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 Go
• AWS SDK for Java
• AWS SDK for Ruby V2
ServicId

Contents

AccountId

Type: String

Required: No

Name

Type: String

Required: No

Names

Type: Array of strings

Required: No

Type

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
ServiceStatistics

Response statistics for a service.

Contents

ErrorStatistics

Information about requests that failed with a 4xx Client Error status code.

Type: ErrorStatistics (p. 29) object

Required: No

FaultStatistics

Information about requests that failed with a 5xx Server Error status code.

Type: FaultStatistics (p. 30) object

Required: No

OkCount

The number of requests that completed with a 2xx Success status code.

Type: Long

Required: No

TotalCount

The total number of completed requests.

Type: Long

Required: No

TotalResponseTime

The aggregate response time of completed requests.

Type: Double

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
TelemetryRecord

Contents

BackendConnectionErrors

  Type: BackendConnectionErrors (p. 25) object
  Required: No

SegmentsReceivedCount

  Type: Integer
  Required: No

SegmentsRejectedCount

  Type: Integer
  Required: No

SegmentsSentCount

  Type: Integer
  Required: No

SegmentsSpilloverCount

  Type: Integer
  Required: No

Timestamp

  Type: Timestamp
  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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Trace

A collection of segment documents with matching trace IDs.

Contents

Duration

The length of time in seconds between the start time of the root segment and the end time of the last segment that completed.

Type: Double

Required: No

Id

The unique identifier for the request that generated the trace’s segments and subsegments.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 35.

Required: No

Segments

Segment documents for the segments and subsegments that comprise the trace.

Type: Array of Segment (p. 33) objects

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
TraceSummary

Metadata generated from the segment documents in a trace.

Contents

Annotations

Annotations from the trace's segment documents.

Type: String to array of ValueWithServiceIds (p. 45) objects map

Required: No

Duration

The length of time in seconds between the start time of the root segment and the end time of the last segment that completed.

Type: Double

Required: No

HasError

One or more of the segment documents has a 400 series error.

Type: Boolean

Required: No

HasFault

One or more of the segment documents has a 500 series error.

Type: Boolean

Required: No

HasThrottle

One or more of the segment documents has a 429 throttling error.

Type: Boolean

Required: No

Http

Information about the HTTP request served by the trace.

Type: Http (p. 32) object

Required: No

Id

The unique identifier for the request that generated the trace's segments and subsegments.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 35.

Required: No
IsPartial

One or more of the segment documents is in progress.

Type: Boolean

Required: No

ResponseTime

The length of time in seconds between the start and end times of the root segment. If the service performs work asynchronously, the response time measures the time before the response is sent to the user, while the duration measures the amount of time before the last traced activity completes.

Type: Double

Required: No

ServiceIds

Service IDs from the trace's segment documents.

Type: Array of ServiceId (p. 37) objects

Required: No

Users

Users from the trace's segment documents.

Type: Array of TraceUser (p. 43) objects

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
TraceUser

Information about a user recorded in segment documents.

Contents

ServiceIds

Services that the user's request hit.

Type: Array of ServiceId (p. 37) objects

Required: No

UserName

The user's name.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
UnprocessedTraceSegment

Information about a segment that failed processing.

Contents

ErrorCode

The error that caused processing to fail.

Type: String
Required: No

Id

The segment's ID.

Type: String
Required: No

Message

The error message.

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
ValueWithServiceIds

Information about a segment annotation.

Contents

AnnotationValue

Values of the annotation.

Type: AnnotationValue (p. 24) object

Required: No

ServiceIds

Services to which the annotation applies.

Type: Array of Serviceld (p. 37) objects

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 Go
- AWS SDK for Java
- AWS SDK for Ruby V2
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 Signature Version 4 Signing Process in the Amazon Web Services General Reference.

**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 Task 2: Create a String to Sign for Signature Version 4 in the Amazon Web Services General Reference.**

**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 Handling Dates in Signature Version 4 in the Amazon Web Services General Reference.

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 Security Token Service, go to AWS Services That Work with IAM in the IAM User Guide.

Condition: If you're using temporary security credentials from the AWS Security Token Service, 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 Task 1: Create a Canonical Request For Signature Version 4 in the Amazon Web Services General Reference.

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: 400

**IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

**InternalFailure**

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

HTTP Status Code: 500

**InvalidAction**

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

HTTP Status Code: 400

**InvalidClientTokenId**

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

HTTP Status Code: 403

**InvalidParameterCombination**

Parameters that must not be used together were used together.

HTTP Status Code: 400

**InvalidParameterValue**

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

**InvalidQueryParameter**

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

**MalformedQueryString**

The query string contains a syntax error.

HTTP Status Code: 404

**MissingAction**

The request is missing an action or a required parameter.

HTTP Status Code: 400
**MissingAuthenticationToken**

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

**MissingParameter**

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

**OptInRequired**

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

HTTP Status Code: 403

**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

**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

**ValidationError**

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

HTTP Status Code: 400