



LA DataStream 1 Configuration API v1

Create and manage DataStream 1 configurations to send logs about edge request-response cycles to selected destinations.

Learn more:

- [DataStream](#)
- Download this API's [RAML and JSON schema descriptors](#).

Overview

Akamai's infrastructure is constantly gathering log entries from thousands of edge servers around the world. You can use the DataStream 1 Configuration API to capture these logs and deliver them to the destination of your choice at low latency.

The DataStream 1 Configuration API lets you create, update, delete, and view data streams for your properties to provide low latency streaming of data in raw and aggregated forms. You can configure a data stream to deliver log data to one or more destinations or integrate it with third-party web services or your own solution to access the data through the [DataStream 1 Pull API](#).

DataStream 1 is an End of Sale (EOS) product and will discontinue on on May 31, 2022. See the [DataStream 1 Migration Guidance](#) for details on DataStream 2 and how to migrate.

Who should use this API

This API offers a programmatic alternative to the features of DataStream 1 available in [Akamai Control Center](#).

Use this API service to monitor request-response cycles delivered through the Akamai platform. Either send the raw or aggregated log data to a destination or fetch it with the DataStream 1 Pull API.

You can use log data for:

- **Monitoring.** You can configure your data streams to send raw logs every 30 seconds or aggregated logs as often as every minute. This low latency streaming can help you proactively monitor performance, detect, and quickly resolve performance degradations. You can use a preferred log analytics platform to continuously ingest DataStream logs and set up real-time dashboards or alerts. This helps you proactively mitigate connectivity problems, service disruptions, and configuration-tuning complications as well as minimize your mean time to recovery (MTTR).
- **Usage tracking.** You can monitor for usage spikes to avoid exceeding commits.
- **Diagnostics.** You can receive pre-aggregated metrics over a specific window of time. Use this option to easily switch between aggregated and raw data views for diagnostics or root cause analysis. For example, you can have your aggregated stream always on for a continuous, high-level view of your CDN health. If the aggregated logs identify high error counts or high average origin response times, you can turn on a raw logs stream for root cause analysis and diagnostics.
- **Troubleshooting.** After you've detected an issue, you may need the necessary data to investigate and isolate the root cause. The log data you receive can help enable long-term trend analysis.

Get started

To configure this API for the first time:

- Contact your Akamai representative to enable the DataStream module for your account.
- Review [Get started with APIs](#) for details on how to set up client tokens to access any Akamai API. These tokens appear as custom hostnames that look like this:
`https://akzz-XXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXX.luna.akamaiapis.net .`
- To enable this API, choose the API service called **Datastream1** and set the access level to **READ-WRITE**.
- The API limits the scope of the returned data by the user's account, its permissions, and access to business objects that are associated with the user and account.

API concepts

This section provides a road map of all the conceptual objects you come across when interacting with the DataStream 1 Configuration API and provides pointers to where you can learn more.

- **Groups.** Each account features a hierarchy of groups, which control access to properties. Using either Control Center or the [Identity Management: User Administration API](#), account administrators can assign properties to specific groups, each with its own set of users and accompanying roles. Your access to any given property depends on the role set for you in its group. Typically, you need a group identifier to create a data stream configuration. See the [Group](#) object.
- **Contracts.** Each account features one or more contracts, each of which has a fixed term of service. You need a contract identifier to create a data stream configuration. Typically, you need a contract identifier to create a data stream configuration.
- **Property.** A property, also referred to as a configuration, provides the main way to control how edge servers respond to various kinds of requests for those assets. Properties apply rules to a set of hostnames, and you can only apply one property at a time to any given hostname. Data streams let you monitor the traffic served by a property. Also, you can monitor one property in up to 15 data streams. See the [Property](#) object.
- **Connector.** A connector, also known as a destination, represents a third-party configuration where you can send the stream's logs. You can configure your data stream to send logs to one or more destinations. DataStream 1 supports these connectors: [Amazon S3](#), [Datadog](#), [Splunk](#), [Sumo Logic](#), [custom HTTPS URL](#), or [DataStream buffer](#).
 - **Push connector.** A connector for which a data stream sends log data to a specific destination for storing, monitoring, and analytical purposes. [Amazon S3](#), [Datadog](#), [Splunk](#), [Sumo Logic](#), and [custom HTTPS URL](#) are available push connectors.
 - **Pull connector.** A connector for which a data stream doesn't send log data to any specific destination but makes it available through the [DataStream 1 Pull API](#) for up to 12 hours. You can configure only one pull connector per stream. [DataStream buffer](#) is the only available pull connector.
- **Stream.** Collects logs about edge request-response cycles for one or more properties. For each defined connector in its configuration, it either streams raw data or aggregated metrics to a destination or makes the log data available through the [DataStream 1 Pull API](#). It lets you control data set parameters you monitor in your logs and delivery conditions for these logs. You can update a data stream through versioning. In a data stream, you can monitor up to 15 properties. See the [Stream configuration](#) and [Stream version](#) objects.

- **Data sets.** Groups parameters that you can either monitor in each request-response cycle or use as aggregation metrics for your logs. Selecting data set parameters lets you control the request-response cycle data that your data stream collects and sends to a destination or stores in a DataStream buffer. See the [Dataset](#) object. These are available data set types:
 - **RAW** data sets provide information about individual edge request-response cycles on the Akamai platform. You can use raw data logs to find details about specific incidents, search the logs for instances using a specific IP address, or analyze the patterns of multiple attacks.
 - **AGGREGATED** data sets provide information about edge request-response cycles aggregated over selected metrics and a time window. You can use aggregated logs to search for the root cause of an error and monitor the performance, security status, and general behavior of your application.
- **DataStream 1 Pull API.** A [reporting API](#) that lets you access log data collected by your data streams for up to 12 hours. You can only use this API to fetch log data for streams with a configured [DataStream buffer](#) connector.

Version management

Every time you edit a data stream, you create a version with an incremented version number. This lets you quickly adapt your existing data streams to collect logs for different properties, modify data set parameters they monitor, or change destinations where they send logs.

Each version that you activate becomes a default version for a data stream with the same activation status as its base version. For example, when you edit version 1 of an active data stream, you create an active version 2 that replaces the previous version on the production network. Conversely, by editing version 1 of an inactive data stream, you create an inactive version 2 of this data stream that replaces the previous version.

You can view and compare all configuration versions within a data stream, but you can only manage the activation status of the latest version. You also can't revert a data stream to any previous version.

Workflow

Follow this basic workflow to configure a data stream:

1. Determine the properties within the selected product that you want to monitor in the data stream. See [List properties](#).

2. Make sure your properties are active on the production network. See [Activate a property in Property Manager](#). Note that you can only collect logs for active properties.
3. Build one or more connector objects for the `connectors` array in the data stream configuration. These are the destinations where you want the data stream to deliver or store logs. See Connector in [API concepts](#).
4. Get and store the data set parameters that you want the data stream to collect. See [List data set parameters](#).
5. Create and configure a data stream. See [Create a stream](#).
6. Enable the DataStream behavior in each property that's part of the data stream. See [DataStream in Property Manager](#). Note that you can only collect logs for DataStream-enabled properties that are active on the production network. You can also further control how your data stream collects logs in Property Manager. For each property, you can specify the streams that you want to receive logs from or configure the behavior in a custom rule to log requests matching your criteria. See [Criteria in Property Manager](#).

Rate limiting

The DataStream 1 Configuration API limits each client to 20 requests per minute. Exceeding this limit results in a 429 error response. Consider this when calling successive operations as part of a loop.

All responses specify these rate limit headers:

- `X-RateLimit-Limit`: The maximum number of tokens allowed.
- `X-RateLimit-Remaining`: The number of tokens remaining. Except for any subsequent requests that reduce the number, this gradually increments until it reaches the `X-RateLimit-Limit`.
- `X-RateLimit-Next`: If the `X-RateLimit-Remaining` has reached 0, this ISO 8601 timestamp indicates when you can next make an additional request.

Resources

This section provides details on the DataStream 1 Configuration API's various operations and parameters.

API summary

Download the [RAML descriptors](#) for this API.

Operation	Method	Endpoint
List properties	GET	/datastream-config-api/v1/datastream1/properties/group/{groupId}
List data set parameters	GET	/datastream-config-api/v1/datastream1/datasets/{type}
List groups	GET	/datastream-config-api/v1/datastream1/groups
List streams	GET	/datastream-config-api/v1/datastream1/streams{?groupId,streamStatus}
Create a stream	POST	/datastream-config-api/v1/datastream1/streams
View a stream version	GET	/datastream-config-api/v1/datastream1/streams/{streamId}{?version}
Edit a stream	PUT	/datastream-config-api/v1/datastream1/streams/{streamId}
Delete a stream	DELETE	/datastream-config-api/v1/datastream1/streams/{streamId}
Activate a stream	PUT	/datastream-config-api/v1/datastream1/streams/{streamId}/activate
Deactivate a stream	PUT	/datastream-config-api/v1/datastream1/streams/{streamId}/deactivate

List properties

Returns properties that are available within the group.

[Request](#) [Parameters](#) [Response](#) [Steps](#)

GET /datastream-config-api/v1/datastream1/properties/group/{groupId}

Sample: /datastream-config-api/v1/datastream1/properties/group/12345

List data set parameters

Returns groups of data set parameters available. You can select the parameters that you want to monitor in the logs collected by your data stream configuration. See the [Create a stream](#) or [Edit a stream](#) operations.

<u>Request</u>	<u>Parameters</u>	<u>Response</u>
----------------	-------------------	-----------------

GET /datastream-config-api/v1/datastream1/datasets/{type}

Sample: /datastream-config-api/v1/datastream1/datasets/RAW

List groups

Returns groups within the context of your account that let you view and create data stream configurations in properties that they have access to.

<u>Request</u>	<u>Response</u>
----------------	-----------------

GET /datastream-config-api/v1/datastream1/groups

List streams

Returns the latest versions of the data stream configurations for all groups within in your account. If you provide the `groupId` as a query parameter, the operation returns the latest versions of the data streams in the group.

Request

Parameters

Response

Steps

GET `/datastream-config-api/v1/datastream1/streams{?groupId,streamStatus}`

Sample: `/datastream-config-api/v1/datastream1/streams?groupId=6483&streamStatus=ACTIVATED`

Create a stream

Creates a data stream configuration. Within a stream configuration, you can select properties to monitor, data set parameters to deliver in logs, and one or more destinations where to send these logs. You can also configure your stream to store your logs and make them available through the DataStream 1 Pull API. See [Connector](#) in [API concepts](#).

Request

Response

Steps

POST `/datastream-config-api/v1/datastream1/streams`

Content-Type: `application/json`

Object type: [StreamConfiguration](#)

Download schema: [SaveStreamRequest.json](#)

Request body:


```
{
  "streamName": "Aggregated logs stream",
  "streamType": "AGGREGATE",
  "groupId": 20628,
  "contractId": "1-GNLXD",
  "timeFrame": 15,
  "propertyIds": [
    472787
  ],
  "datasetFieldIds": [
    86,
    79,
    80,
    81,
    82
  ],
  "connectors": [
```

View a stream version

Returns a version of a data stream, including details about its monitored properties, logged data set parameters, and delivery destinations. If you omit the `version` query parameter, this operation returns the latest version of the data stream.

Request

Parameters

Response

Steps

GET /datastream-config-api/v1/datastream1/streams/{streamId}{?version}

Sample: /datastream-config-api/v1/datastream1/streams/6301?version=1

Edit a stream

Updates the latest version of a data stream. Running this operation creates a version a data stream that becomes a default version for this stream. Note that only active streams collect and send logs to their destinations. See [Version management](#) and [Activate a stream](#).

[Request](#)[Parameters](#)[Response](#)[Steps](#)

PUT /datastream-config-api/v1/datastream1/streams/{streamId}

Sample: /datastream-config-api/v1/datastream1/streams/6301

Content-Type: application/json

Object type: [StreamConfiguration](#)

Download schema: [EditStreamRequest.json](#)

Request body:

```
{
  "streamName": "trial",
  "streamType": "RAW",
  "groupId": 158842,
  "contractId": "1-FE6JR",
  "sampleRate": 48,
  "timeFrame": null,
  "propertyIds": [
    180934
  ],
  "datasetFieldIds": [
    65,
    66,
    67,
    68,
    69,
    70,
```

Delete a stream

Deletes a stream. Deleting a stream means that you won't be able to activate this data stream again, and that you'll stop receiving logs for the properties that this data stream monitors. To delete a data stream, you need to run the [Deactivate a stream](#) operation.

[Request](#)[Parameters](#)[Response](#)[Steps](#)

DELETE /datastream-config-api/v1/datastream1/streams/{streamId}

Sample: /datastream-config-api/v1/datastream1/streams/6301

Activate a stream

Activates the latest version of a data stream. Activating a data stream takes about 15 minutes. Once a data stream is active, it starts collecting and sending logs to its destinations. If you want to stop receiving these logs, you can deactivate a data stream at any time. See the [Deactivate a stream](#) operation.

<u>Request</u>	<u>Parameters</u>	<u>Response</u>	<u>Steps</u>
----------------	-------------------	-----------------	--------------

PUT /datastream-config-api/v1/datastream1/streams/{streamId}/activate

Sample: /datastream-config-api/v1/datastream1/streams/6301/activate

Deactivate a stream

Deactivates the latest version of a stream. Deactivating a stream means that you stop receiving logs for the properties that this data stream monitors. Deactivating a data stream takes approximately 15 minutes. If you want to start receiving these logs again, you can activate this data stream at any time. See the [Activate a stream](#) operation.

<u>Request</u>	<u>Parameters</u>	<u>Response</u>	<u>Steps</u>
----------------	-------------------	-----------------	--------------

PUT /datastream-config-api/v1/datastream1/streams/{streamId}/deactivate

Sample: /datastream-config-api/v1/datastream1/streams/6301/deactivate

Data

This section provides you with the data model for the DataStream 1 Configuration API.

Download the [JSON schemas](#) for this API.

This section's data schema tables list membership requirements as follows:

- ✓ Member is required in requests, or always present in responses, even if its value is empty or null.
- Member is optional, and may be omitted in some cases.
- ✗ Member is out of scope, and irrelevant to the specified interaction context. If you include the member in that context, it either triggers an error, or is ignored.

Property

Provides details about a property within the context of the group.

Download schema: [Property.json](#)

Sample [GET response](#):

```
{
  "propertyId": 382631,
  "propertyName": "property.com"
}
```

Property members

Member	Type	Required	Description
Property: Provides details about a property within the context of the group.			
propertyId	Integer	○	The identifier of the property.

Member	Type	Required	Description
propertyName	String	○	The descriptive label for the property.

Group

Provides detailed information about the hierarchical context of the group. It also informs you whether you can view and create data stream configurations within this group.

Download schema: [Group.json](#)

Sample [GET response](#):

```
{
  "parentGroupId": 20628,
  "groupId": 49851,
  "groupName": "Performance Group",
  "description": null,
  "accountId": "1-BGIGR",
  "enabled": true,
  "contractIds": [
    "1-GNLXD"
  ],
  "childGroupIds": [
    98579,
    53491,
    130231,
    58588
  ]
}
```

Group members

Member	Type	Required	Description
accountId	String	○	Identifies the account that the group is part of.
childGroupIds	Array	○	Identifies the child groups within the group.

Member	Type	Required	Description
contractIds	Array	○	Identifies the contracts associated with the group.
description	String, Null	○	Describes the group.
enabled	Boolean	○	Whether this group allows you to create and view data stream configurations.
groupId	Integer	○	Identifies the group.
groupName	String	○	Identifies the name of the group.
parentGroupId	Integer	○	Identifies the parent group.

Dataset

Provides information about a group of data set parameters available.

Download schema: [Dataset.json](#)

Sample [GET response](#):

```
{
  "datasetGroupName": "network",
  "datasetGroupDescription": "Network Data",
  "datasetFields": [
    {
      "datasetFieldId": 72,
      "datasetFieldName": "nw",
      "datasetFieldDescription": "Network"
    },
    {
      "datasetFieldId": 73,
      "datasetFieldName": "nwType",
      "datasetFieldDescription": "Network Type"
    },
    {
      "datasetFieldId": 74,
      "datasetFieldName": "throughput",
```

Dataset members

Member	Type	Description
Dataset: Provides information about a group of data set parameters available.		
datasetFields	<u>Dataset</u> . <u>datasetFields[]</u>	A list of data set parameters available within the data set group.
datasetGroup Description	String	Describes the dataset group.
datasetGroupName	String	A name of the dataset group.
Dataset.datasetFields[]: A list of data set parameters available within the data set group.		
datasetField Description	String	Describes the data set field.
datasetFieldId	Integer	Identifies the field.
datasetFieldName	String	A name of the data set field.

StreamVersion

Provides detailed information about the latest configuration version of a data stream. It provides selected data set parameters, associated property identifiers, destination details, and version status.

Download schema: [StreamDetail.json](#)

Sample [GET response](#):

```
{
  "streamId": 10187,
  "streamName": "Example Stream",
  "streamVersionId": 23,
  "createdBy": "user",
  "createdDate": "30-07-2020 07:22:40 GMT",
  "activationStatus": "ACTIVATED",
  "groupId": 20628,
  "groupName": "Group Name",
  "contractId": "1-GNLXD",
  "streamType": "AGGREGATE",
  "modifiedBy": "user",
  "modifiedDate": "02-08-2020 08:23:31 GMT",
  "timeFrame": 1,
  "datasets": [
    {
      "datasetGroupName": "edge_response_time",
```

StreamVersion members

Member	Type	Description
		<p>StreamVersion: Provides detailed information about the latest configuration version of a data stream. It provides selected data set parameters, associated property identifiers, destination details, and version status.</p>
activationStatus	Enumeration	The activation status of the data stream configuration version. These are possible values: <code>ACTIVATING</code> , <code>ACTIVATED</code> , <code>DEACTIVATING</code> , or <code>DEACTIVATED</code> . See the Activate a stream and Deactivate a stream operations.
connectors	Array of these objects: Amazon S3 , Https , Splunk , SumoLogic , Datadog , or Data Stream buffer	Provides detailed information about connector configurations in the stream when viewing a stream's version or the connector names when listing available streams .
contractId	String	Identifies the contract that you created the stream for.
createdBy	String	The username who created the stream.

Member	Type	Description
createdDate	String	The date and time when the stream was created.
datasets	<u>StreamVersion.datasets[]</u>	A list of data set parameters selected from the associated template that the stream monitors in logs.
groupId	Integer	Identifies the group that has access to the product and that you created the stream configuration for.
groupName	String	The name of the user group that you created the stream for.
modifiedBy	String	The username who modified the stream.
modifiedDate	String	The date and time when the stream was modified.
properties	<u>StreamVersion.properties[]</u>	Identifies the properties that you monitor in the stream.
streamId	Integer	Identifies the stream.
streamName	String	The name of the stream.
streamType	Enumeration	Specifies the type of stream, either RAW or AGGREGATE.
streamVersionId	Integer	Identifies the configuration version of the stream.
StreamVersion.datasets []: A list of data set parameters selected from the associated template that the stream monitors in logs.		
datasetFields	<u>StreamVersion.datasets[].datasetFields[]</u>	A list of data set parameters in the group that the stream monitors.
datasetGroup Description	String	A descriptive label for the dataset group.

Member	Type	Description
datasetGroupName	String	The name of the dataset group.
StreamVersion.datasets[].datasetFields[]: A list of data set parameters in the group that the stream monitors.		
datasetField Description	String	Describes the data set field.
datasetFieldId	Integer	Identifies the field.
datasetFieldName	String	A name of the data set field.
StreamVersion.properties[]: Identifies the properties that you monitor in the stream.		
propertyId	Integer	The identifier of the property.
propertyName	String	The descriptive label for the property.

StreamConfiguration

Provides information that you need to specify when creating a data stream configuration.

Download schema: [SaveStreamRequest.json](#) , [EditStreamRequest.json](#)

Sample [POST request](#):

```
{
  "streamName": "Aggregated logs stream",
  "streamType": "AGGREGATE",
  "groupId": 20628,
  "contractId": "1-GNLXD",
  "timeFrame": 15,
  "propertyIds": [
    472787
  ],
  "datasetFieldIds": [
    86,
    79,
    80,
    81,
    82
  ],
  "connectors": [
```

StreamConfiguration members

Member	Type	POST	PUT	Description
StreamConfiguration: Provides information that you need to specify when creating a data stream configuration.				
connectors	Array of these objects: Amazon S3 , Https , Splunk , SumoLogic , Datadog , or DataStream buffer	✓	○	Specifies one or more destinations where the stream sends or stores log files. When creating a stream, you need to configure at least one connector object. When editing a stream, you can omit this member to leave the existing connector configurations unchanged. See API concepts for the set of available connectors.
contractId	String	✓	✓	Identifies the contract.
datasetFieldIds	Array	✓	✓	Identifiers of the data set parameters that you want to monitor in logs.

Member	Type	POST	PUT	Description
groupId	Integer	✓	○	Identifies the group that has access to this data stream's configuration.
propertyIds	Array	✓	✓	Identifies the properties that you want to monitor in the stream. Note that a stream can only log data for active properties. You can activate a property in Property Manager in Property Manager.
sampleRate	Integer	○	○	Specifies the sampling percentage of data to send to your destination for the RAW stream type. By default, it's 25.
streamName	String	✓	○	The name of the stream.
streamType	Enumeration	✓	○	The type of stream that you want to create, either RAW or AGGREGATE.
timeFrame	String	○	○	Specifies how often you want the system to send the log file to your destination for the AGGREGATE stream type. These are possible values: 1, 5, 15, 30, or 60 minutes. By default, it's 1.

StreamUpdate

Provides information about a data stream that you created or edited. It also provides information about streams that share properties with the created or edited configuration.

Download schema: [SaveStreamUpdate.json](#)

Sample [POST response](#):

```

{
  "streamVersionKey": {
    "streamId": 6286,
    "streamVersionId": 1
  },
  "affectedStreams": [
    {
      "streamName": "CDN Health Stream",
      "streamVersionKey": {
        "streamId": 6286,
        "streamVersionId": 1
      }
    },
    {
      "streamName": "Aggregated Stream",
      "streamVersionKey": {
        "streamId": 6285,

```

StreamUpdate members

Member	Type	Description
StreamUpdate		StreamUpdate: Provides information about a data stream that you created or edited. It also provides information about streams that share properties with the created or edited configuration.
affectedProperties	StreamUpdate.affectedProperties[]	The set of properties that the affectedStreams share with the created or edited configuration.
affectedStreams	StreamUpdate.affectedStreams[]	The set of streams that share properties with the created or edited configuration.
streamVersionKey	StreamUpdate.streamVersionKey	Provides the stream and version identifiers of the created or edited configuration.
StreamUpdate.affectedProperties []: The set of properties that the affectedStreams share with the created or edited configuration.		
propertyId	Integer	Identifies the shared property.
propertyName	String	Identifies the property's name.

Member	Type	Description
StreamUpdate.affectedStreams []: The set of streams that share properties with the created or edited configuration.		
streamName	String	Identifies the stream's name.
streamVersionKey	<u>StreamUpdate.affectedStreams[].streamVersionKey</u>	Provides the stream and version identifiers of the configuration that shares properties with the created or edited configuration.
StreamUpdate.affectedStreams [].streamVersionKey: Provides the stream and version identifiers of the configuration that shares properties with the created or edited configuration.		
streamId	Integer	Identifies the stream that shares properties with the created or edited configuration.
streamVersionId	Integer	Identifies the latest version of the stream.
StreamUpdate.streamVersionKey: Provides the stream and version identifiers of the created or edited configuration.		
streamId	Integer	Identifies the stream that you created or edited.
streamVersionId	Integer	Identifies the latest version of the stream that you created or edited.

AmazonS3

Provides details about the Amazon S3 connector in a data stream. When you create or edit a stream configuration, DataStream validates this connector type. As part of this validation process, it uses the provided `accessKey` and `secretAccessKey` values to save an `akamai_write_test_2147483647.txt` file in your S3 folder. You can only see this file if you've write access to the Amazon S3 bucket and folder that you're trying to send logs to, and if the entire validation process is successful.

Download schema: [S3.json](#) , [S3SavedConnector.json](#)

Sample [POST request](#):

```
{
  "path": "log",
  "connectorName": "S3_connector",
  "bucket": "media-datastream.akamai.com",
  "region": "AP_Mumbai",
  "accessKey": "AKIA6DK7TDQLVGZ3TYP1",
  "secretAccessKey": "1T2ll1H4dXWx5itGhpc7FlSbvv0vky1098nTtEMg",
  "connectorType": "S3"
}
```

AmazonS3 members

Member	Type	Required	Description
<p>AmazonS3: Provides details about the Amazon S3 connector in a data stream. When you create or edit a stream configuration, DataStream validates this connector type. As part of this validation process, it uses the provided accessKey and secretAccessKey values to save an akamai_write_test_2147483647.txt file in your S3 folder. You can only see this file if you've write access to the Amazon S3 bucket and folder that you're trying to send logs to, and if the entire validation process is successful.</p>			
accessKey	String	○	The access key identifier that you use to authenticate requests to your S3 account. See Managing access keys in AWS API .
bucket	String	○	The name of the S3 bucket. See Working with Amazon S3 buckets in AWS .
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.
connectorType	Enumeration	○	The name of the connector type. Set it to S3 for this connector type.
path	String	○	The folder path associated with your S3 bucket where you want to store your logs.

Member	Type	Required	Description
region	String	○	The name of the AWS region where your S3 bucket resides. See Regions, availability zones, and local zones in AWS .
secretAccessKey	String	○	The secret access key identifier that you use to authenticate requests to your S3 account.

Datadog

Provides details about the Datadog connector configuration in a data stream.

Download schema: [Datadog.json](#), [DatadogSavedConnector.json](#)

Sample [POST request](#):

```
{
  "service": "datastream-config-api",
  "authToken": "6fe69bf3791009bd7b3bv18dc2c68fe",
  "connectorName": "Datadog_connector",
  "url": "https://http-intake.logs.datadoghq.com/v1/input/",
  "connectorType": "DATADOG"
}
```

Datadog members

Member	Type	Required	Description
Datadog: Provides details about the Datadog connector configuration in a data stream.			
authToken	String	○	The API key associated with your Datadog account. See View API keys in Datadog .
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.

Member	Type	Required	Description
connectorType	Enumeration	○	The name of the connector type. Set it to DATADOG for this connector type.
service	String	○	The service of the Datadog connector. See View Datadog services list .
url	String	○	The Datadog URL where you want to store logs.

Splunk

Provides detailed information about Splunk connector that you can use in your stream.

Download schema: [Splunk.json](#), [SplunkSavedConnector.json](#)

Sample [POST request](#):

```
{
  "connectorName": "Splunk_connector",
  "url": "https://http-intake.logs.splunk.com/v1/input/",
  "authToken": "244fd617-51c5-4b2e-916d-54fb-af62",
  "connectorType": "SPLUNK"
}
```

Splunk members

Member	Type	Required	Description
Splunk: Provides detailed information about Splunk connector that you can use in your stream.			
authToken	String	○	The API key associated with your Splunk account. See View usage of Auth tokens in Splunk .
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.

Member	Type	Required	Description
connectorType	Enumeration	○	The name of the connector type. Set this value to SPLUNK for this connector type.
url	String	○	The Splunk URL where you want to store your logs.

SumoLogic

Provides detailed information about a Sumo Logic connector that you can use in your stream.

Download schema: [SumoLogic.json](#) , [SumoLogicSavedConnector.json](#)

Sample [POST request](#):

```
{
  "connectorName": "Sumo_connector",
  "url": "https://endpoint2.collection.us2.sumologic.com/receiver/v1/http/ZaVnC4dhaV3Z6XTKv2zgi6YdjR2a9ZpU7GhuVgWxr3xod1eGTAztg6t_8fLbxakler2gm_l7FepmaQFj-Nc9_GLJIItWCJgFkVfFSF75x_LHzVZSHIkQ4xQ==",
  "connectorType": "SUMO_LOGIC"
}
```

SumoLogic members

Member	Type	Required	Description
SumoLogic: Provides detailed information about a Sumo Logic connector that you can use in your stream.			
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.
connectorType	Enumeration	○	The name of the connector type. Set this value to SUMO_LOGIC for this connector type.
url	String	○	The Sumo Logic URL where you want to store your logs.

Https

Provides detailed information about a custom HTTPS connector that you can use in your stream.

Download schema: [HTTPS.json](#), [HTTPSSavedConnector.json](#)

Sample [POST request](#):

```
{
  "connectorName": "Https_connector",
  "url": "https://endpoint2.collection.us2.sumologic.com/receiver/v1/http/ZaVnC4dhaV3Z6XTKv2zgi6YdjR2a9ZpU7GhuVgWxr3xod1eGTAztg6t_8fLbxakler2gm_l7FepmaQFj-Nc9_GLJIItWCJgFkVfFSF75x_LHzVZSHIkQ4xQ==",
  "connectorType": "HTTPS"
}
```

Https members

Member	Type	Required	Description
Https: Provides detailed information about a custom HTTPS connector that you can use in your stream.			
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.
connectorType	Enumeration	○	The name of the connector type. Set this value to HTTPS for this connector type.
url	String	○	The secure URL where you want to store your logs.

DataStreamBuffer

Provides detailed information about the DataStream buffer connector that you can use in a data stream. To start using the DataStream Pull API to fetch your logs from the DataStream buffer configured for a stream, you need to wait around 10 minutes after activating the stream on the production network. Note that you can configure only one DataStream buffer connector for a stream. See [Connector in API concepts](#).

Download schema: [DatastreamBuffer.json](#) , [DatastreamBufferSavedConnector.json](#)

Sample [POST request](#):

```
{
  "connectorName": "Datastream Buffer",
  "connectorType": "DATASTREAM_BUFFER"
}
```

DataStreamBuffer members

Member	Type	Required	Description
<p>DataStreamBuffer: Provides detailed information about the DataStream buffer connector that you can use in a data stream. To start using the DataStream Pull API to fetch your logs from the DataStream buffer configured for a stream, you need to wait around 10 minutes after activating the stream on the production network. Note that you can configure only one DataStream buffer connector for a stream. See Connector in API concepts.</p>			
connectorId	Integer	○	Identifies the connector associated with the data stream.
connectorName	String	○	The name of the connector.
connectorType	Enumeration	○	The name of the connector type. Set this value to DATASTREAM_BUFFER for this connector type.

Errors

This section provides details on the data object that reflect the API's common response to error cases, and lists the API's range of response status codes for both error and success cases.

Error responses

In error cases, the API responds with JSON objects that follow the [HTTP Problem Details](#). This sample shows a bad request error, where the `title` is a descriptive label for the overall problem, and the `instance` may be useful if you need to communicate the problem to your Akamai support representative. It also includes an optional `errors` array that lists potentially more than one problem detected in the request.

```
{
  "type": "bad-request",
  "title": "Erroneous data input",
  "instance": "8ed959ae-bc22-43f4-893c-2f293518f258",
  "status": 400,
  "errors": [
    {
      "type": "bad-request",
      "title": "Bad Request",
      "instance": "1664e1a7-d916-4cf9-944f-2d9f6d176f8f",
      "detail": "Expiry Date of Previous Key is required"
    }
  ]
}
```

HTTP status codes

The API produces these set of HTTP status codes for both success and failure scenarios:

Code	Description
<u>200</u>	The operation succeeded.
<u>201</u>	Resource created.
<u>202</u>	Resource successfully accepted. This is returned on an activation request. It doesn't mean that the activation succeeded, but that the API was able to act upon the request.
<u>400</u>	Bad Request.
<u>401</u>	Unauthorized request.
<u>402</u>	Failed request.
<u>403</u>	Access is forbidden. You don't have access to the requested resource.
<u>404</u>	Resource not found.
<u>405</u>	Method not allowed.
<u>415</u>	Unsupported media type.
<u>422</u>	Unprocessable entity.

Code	Description
429	Too many requests. See Rate limiting .
500	Internal server error. Unexpected error.

COMPANY

[Akamai.com](#)

[Akamai Locations](#)

[Contact Support](#)

DEVELOPER

[Documentation](#)

[Blog](#)

[Resources](#)

STAY IN TOUCH



WYBIERZ JĘZYK ▼