

# AirSync

## Partner API Documentation

### Overview

[API Documentation](#)

[Environments](#)

[Credentials](#)

### API Usage

[Initial Authorization](#)

[General Usage](#)

[Log Confirmation](#)

### Aircraft Subscription

[Establish an Endpoint](#)

[Subscribe](#)

[Unsubscribe](#)

[Examples](#)

### Change Log

## Overview

### API Documentation

[https://service-dev.air-sync.com/partner\\_api/v1/docs](https://service-dev.air-sync.com/partner_api/v1/docs)

### Environments

Sandbox/Development - [https://service-dev.air-sync.com/partner\\_api/v1](https://service-dev.air-sync.com/partner_api/v1)

Production - [https://api.air-sync.com/partner\\_api/v1](https://api.air-sync.com/partner_api/v1)

### Credentials

All partner accounts will receive an API key and secret. The key/secret pair will be different between the sandbox and production environments.

# API Usage

## Initial Authorization

1. Join API key and secret together
  - a. [API Key]:[API Secret] Ex: 12345:ABCDE
2. Create base 64 hash of this new string
3. Call POST /auth/
  - a. Authorization header with value:
    - i. Basic <base 64 hash>
  - b. Response will be your access token:
    - i. {"access\_token": "<token value>"}
4. Use this access token value in all subsequent calls for aircraft and logs
5. Token is valid for 1 day (non-rolling)
6. Repeat 1-3 to obtain a new token when the previous one has expired

Note: We recommend maintaining expiration date and obtaining new token once a day; or, after receiving a 401 permission denied response.

## General Usage

After initial authorization, for all other API method calls to /aircraft and /logs endpoints, include the "Authorization" header with value: Bearer <access token>

## Log Confirmation

After receiving and/or downloading a log file, the partner's system must post confirmation with their corresponding unique database key.

POST /logs/<id>/confirm?partner\_key=<partner\_key>

id = AirSync log ID

partner\_key = Partner's corresponding database key for the log

## Aircraft Subscription

When subscribed to a new aircraft, the partner system will be notified after new logs are received in AirSync. Subscribing to an aircraft is preferable to polling for new logs.

AirSync utilizes Amazon SNS for notifications.

Only one subscription per aircraft per partner is supported.

## Establish an Endpoint

In order to subscribe (and auto-subscribe) to an aircraft, the receiving partner must establish an endpoint capable of receiving and parsing the JSON data structure.

AirSync supports two endpoint protocols:

1. Webhook (https)
2. Email (email\_json)

**IMPORTANT: Your endpoint protocol MUST be active before you initiate a subscribe request. The system cannot unsubscribe without first confirming the subscription.**

## Subscribe

POST /aircraft/<id>/subscribe?protocol=<protocol>&endpoint=<endpoint>

id = AirSync Aircraft ID

protocol = https | email\_json

endpoint = URL or email address (based on protocol)

### Confirming the Subscription

After successful subscription, SNS will post a JSON message to the endpoint. See *below for example*.

#### Webhook (https)

Messages are posted with Content-Type “text/plain”. The raw data payload must be converted to JSON (or another data type supported by your platform).

#### Email (email\_json)

The email specified in the 'endpoint' field will receive an email containing a JSON object in the email body. This text value must be converted to JSON (or another data type supported by your platform).

This confirmation email will contain a JSON attribute "Type" = "SubscriptionConfirmation".

To confirm and activate the subscription, your endpoint must issue a GET request to the value in the 'SubscribeURL' attribute from the JSON object.

## **Receiving Log Notifications**

After confirming the subscription request, the endpoint will post JSON requests for each message published to aircraft. See *below for example*.

### **Webhook ([https](https://))**

Messages are posted with Content-Type “text/plain”. The raw data payload must be converted to JSON (or another data type supported by your platform).

### **Email ([email\\_json](mailto:email_json))**

Messages are included in the email body as plain text. This text value must be converted to JSON (or another data type supported by your platform).

For ongoing aircraft notifications, the 'Type' attribute will be 'Notification' and the request JSON will contain the message content in 'Subject' and 'Message' fields.

AirSync's JSON payload is contained within the Message field of the posted JSON object or email body. It is an escaped string and must be converted. Use the “aircraft\_id” and log “id” attribute values to query and fetch the log in subsequent API requests.

## **Auto-Subscribe**

Each partner that implements the Aircraft Subscription functionality can request AirSync to activate “auto-subscription”. If your system has a common webhook for all aircraft, it can be configured to auto-subscribe and a confirmation automatically posted when a customer enables your service within AirSync.

## **Unsubscribe**

If notifications are no longer desired, the partner system may unsubscribe from the aircraft.

`DELETE /aircraft/<id>/unsubscribe`

`id` = AirSync Aircraft ID

**IMPORTANT:** Do not use the UnsubscribeURL provided in the notification messages. Please unsubscribe using the unsubscribe API method above.

## Examples

Example Subscription Confirmation Payload:

POST

Content-Type: text/plain

```
{ "Type" : "SubscriptionConfirmation",
  "MessageId" : "e0875264-7d53-4b98-9423-21ee1b871eef",
  "Token" : "TOKEN_VALUE",
  "TopicArn" : "arn:aws:sns:us-east-1:1234567890:AirSync-Aircraft-Topic-Name",
  "Message" : "You have chosen to subscribe to the topic
arn:aws:sns:us-east-1:1234567890:AirSync-Aircraft-Topic-Name.\nTo confirm the subscription, visit
the SubscribeURL included in this message.",
  "SubscribeURL" :
  "https://sns.us-east-1.amazonaws.com/?Action=ConfirmSubscription&TopicArn=arn:aws:sns:us-east-1:1
234567890:AirSync-Aircraft-Topic-Name&Token=TOKEN_VALUE",
  "Timestamp" : "2023-03-23T20:31:20.388Z",
  "SignatureVersion" : "1",
  "Signature" : "XXXXXXXXXXXXXX",
  "SigningCertURL" : "https://sns.us-east-1.amazonaws.com/SimpleNotificationService-YYYYYYY.pem" }
```

Example Notification Payload:

POST

Content-Type: text/plain

```
{ "Type" : "Notification",
  "MessageId" : "663e492c-2f7a-59e2-baa3-15a1582281c5",
  "TopicArn" : "arn:aws:sns:us-east-1:1234567890:TAILNUMBER-LogReceived-NNNNN",
  "Subject" : "Log Received",
  "Message" : "{\"time_start\": \"2022-12-24T15:25:00Z\", \"time_end\": \"2022-12-25T16:04:59Z\",
\"id\": 12345678, \"aircraft_id\": 1234, \"origin\": \"BINP\", \"destination\": \"BINP\",
\"num_takeoffs\": 1, \"num_landings\": 1, \"flight_time\": 24.461389, \"hobbs_time\": 24.666389,
\"flight_end\": 5000001.80639, \"hobbs_end\": 5000002.61361, \"distance_flown\": 13000000000.826,
\"fuel_end\": 559.5, \"fuel_used\": 153.8, \"fuel_units\": \"gals\", \"timer_precision\": null,
\"timestamp_uploaded\": \"2022-12-25T16:05:31Z\", \"cycles\": 1.0, \"cycles_end\": 500.0,
\"engine_time\": 0.666389, \"flight_analysis_alert_level\": null}",
  "Timestamp" : "2023-03-16T19:59:20.749Z",
  "SignatureVersion" : "1",
  "Signature" : "XXXXXXXXXXXXXX",
  "SigningCertURL" : "https://sns.us-east-1.amazonaws.com/SimpleNotificationService-YYYYYYY.pem",
  "UnsubscribeURL" :
  "https://sns.us-east-1.amazonaws.com/?Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-east-1:12
34567890:TAILNUMBER-LogReceived-NNNNN:SUBSCRIPTION_ID_VALUE"
}
```

# Change Log

Date	Description	Modified By
2022-12-20	Initial version	Chris Mentch
2023-03-24	Added Aircraft Subscription and Log Confirmation sections	Chris Mentch