

# Event Tokens

- [What is an Event Token?](#)
- [What is signed by the event token?](#)
- [What key and algorithm are used to sign the JWT?](#)
- [Token Generation Example](#)

## What is an Event Token?

Each time Flock either sends an event to the event listener URL, or opens a widget or browser, it includes an event token in the form of a [JSON Web Signature \(JWS\)](#). This is a quick method for verifying that the request really did originate from Flock and prevent spoofing attacks.

## What is signed by the event token?

The JWS signs a [JSON Web Token \(JWT\)](#) with the following attributes:

Attribute Name	Attribute Type	Attribute Description
appId	String	The app's id
userId	String	The user's id
exp	Number	Expiration time of the token
iat	Number	Time at which the token was issued
jti	String	(JWT ID) A unique identifier for the token

The *exp* and *iat* are numeric values representing the number of seconds from 1970-01-01T00:00:00Z UTC until the specified UTC date/time, ignoring leap seconds.

Note that the same event token may be sent more than once.

## What key and algorithm are used to sign the JWT?

The key used to sign the event token is the app secret. The algorithm used is HMAC SHA-256.

## Token Generation Example

The example below demonstrates how an event token is generated. You can consult the [JWS specification](#) for further details about this process. However, for most languages you should be able to find a JWT library with JWS support out of the box, so you wouldn't need to implement the verification and decoding yourself.

Assuming that the following JWT payload needs to be signed:

```
{
  "appId": "my-app",
  "userId": "u:3d004302-a97d-4016-91b4-6c221bb4781d",
  "exp": 1469541580,
  "iat": 1469541572,
  "jti": "568eadf8-77fc-4108-91da-d94da46d709b"
}
```

Further, assuming that the app secret that would be used to sign the payload is 869eb1d0-419d-4747-98b4-6d81360a6681, the resultant HMAC-SHA256 signature (encoded using URL-safe base64) is i\_jz-dXk1UV5SKbcbZVAyMAeHIKDY98YTts0u1-ocVegM.

