API Authentication token

Last Modified on 01.09.2025

Overview

The Authentication Token allows you to prevent unauthorised use of the Userlane snippet by adding authentication and verification steps. When included into Userlane snippet, the token is passed to API during initialization to authenticate users from the first request and provide an additional security layer.

How it works

This token verifies a User at the moment of their **first request** to Userlane, adding an **extra layer of security**. It is in addition to - not a replacement for - the Public ID.

This piece to include into Userlane snippet:

```
Userlane('init', '<property-id>', { clientToken: '...' });
```

Key points:

- Tokens are **optional -** omit if not needed.
  - if activated then each request from assistant to API
    will be accompanied with token
- Token itself is generated & saved by customer (below is described how to do so in Portal), separate per each app
  - If you have multiple apps with deployed Userlane - you cans set up separate token for each
- Use short-lived tokens, rotate regularly.
- Never log or expose tokens in unsecured locations.
- If provided, the token is validated before any User data is processed.

Benefits

Heres' the key benefits the token gives to the customers.

Enhanced security:

In this era of the rapid boom of the external 3rd party AI agents that could hypothetically retrieve sensitive internal info with the requests to the API's.

Better compliance:

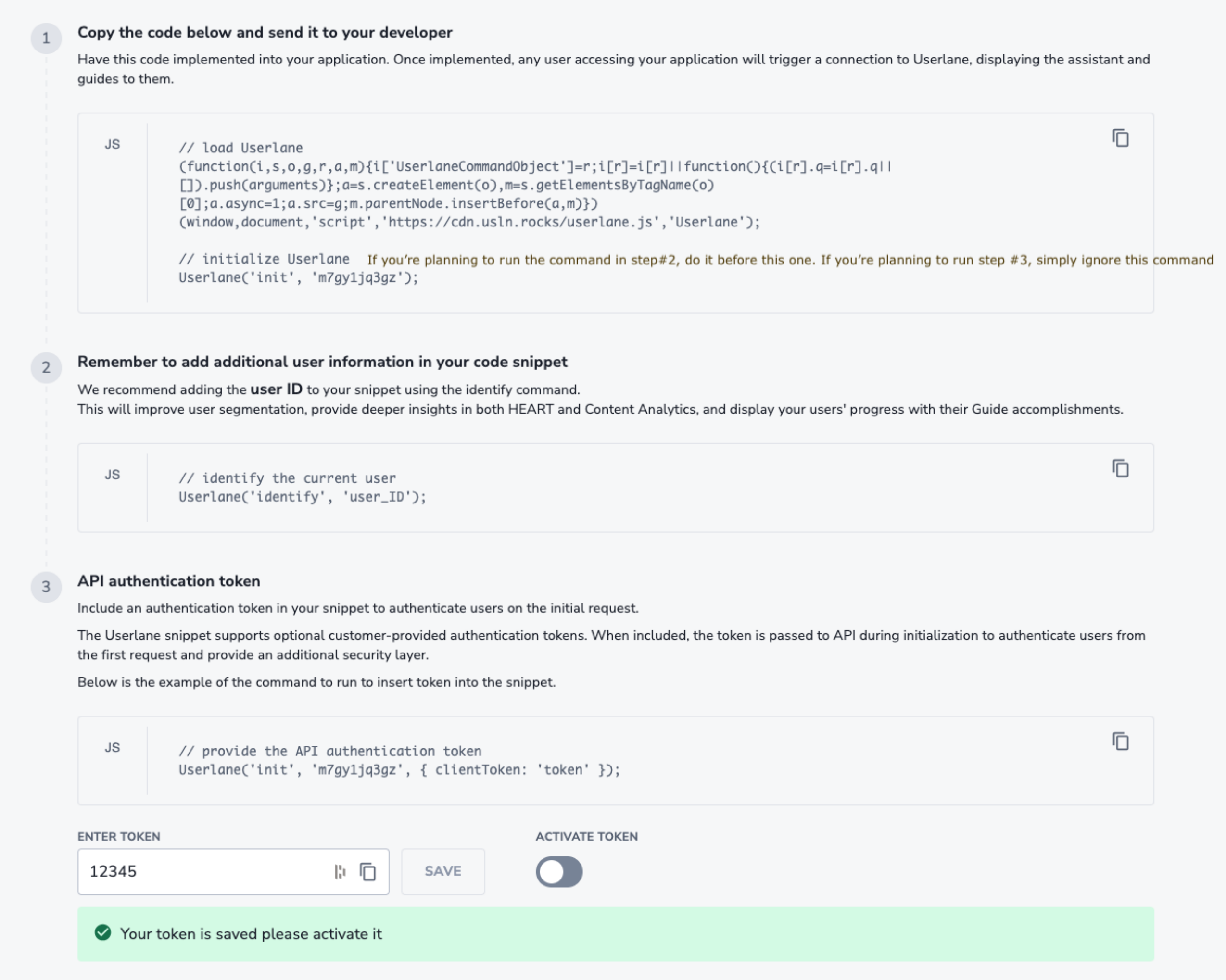
If you're running a business in highly regulated industries which have stricter policies and it could be very useful for you since it adds an additional security layer

How to activate it?

In order to save and activate the authentication token you should have access to the Portal and have Admins rights

Do these steps to get therte:

1. Sign in to Portal
2. Go to **Settings -> Application Settings -> Snippet Setup**
3. **Come up with token, save it**
4. **Then run the command described above to insert saved token into snippet**
5. **Finally, turn the toggle ON**



1 **Copy the code below and send it to your developer**

Have this code implemented into your application. Once implemented, any user accessing your application will trigger a connection to Userlane, displaying the assistant and guides to them.

JS

```
// load Userlane
(function(i,s,o,g,r,a,m){i['UserlaneCommandObject']=r;i[r]=i[r]||function(){(i[r].q=i[r].q||
[]).push(arguments)};a=s.createElement(o),m=s.getElementsByTagName(o)
[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)})
(window,document,'script','https://cdn.usln.rocks/userlane.js','Userlane');

// initialize Userlane
Userlane('init', 'm7gy1jq3gz');
```

If you're planning to run the command in step#2, do it before this one. If you're planning to run step #3, simply ignore this command

2 **Remember to add additional user information in your code snippet**

We recommend adding the **user ID** to your snippet using the identify command.
This will improve user segmentation, provide deeper insights in both HEART and Content Analytics, and display your users' progress with their Guide accomplishments.

JS

```
// identify the current user
Userlane('identify', 'user_ID');
```

3 **API authentication token**

Include an authentication token in your snippet to authenticate users on the initial request.

The Userlane snippet supports optional customer-provided authentication tokens. When included, the token is passed to API during initialization to authenticate users from the first request and provide an additional security layer.

Below is the example of the command to run to insert token into the snippet.

JS

```
// provide the API authentication token
Userlane('init', 'm7gy1jq3gz', { clientToken: 'token' });
```

ENTER TOKEN: 12345

ACTIVATE TOKEN: ○

✓ Your token is saved please activate it