

Blendr.io - SSO example code for SaaS companies

Blendr offers 3 different SSO methods: JWT, Hash and OAuth2. This article focuses on JWT to authenticate the users automatically in Blendr. Note that SaaS companies will embed the Hub (Blendr's marketplace with templates) inside their own solution. A common requirement is to get the user automatically logged in, thus avoiding a second manual login with separate credentials.

Before you continue with this article please make sure that SSO is activated in your Blendr tenant. Otherwise, this SSO example code won't work for you.



Activate SSO: Please contact support, in order to activate SSO for your Hub. This must be done before you can start testing the SSO integration.

Source: <https://help.qlik.com/en-US/blendr/Content/documentation/sso-integration-for-saas-partners.htm>

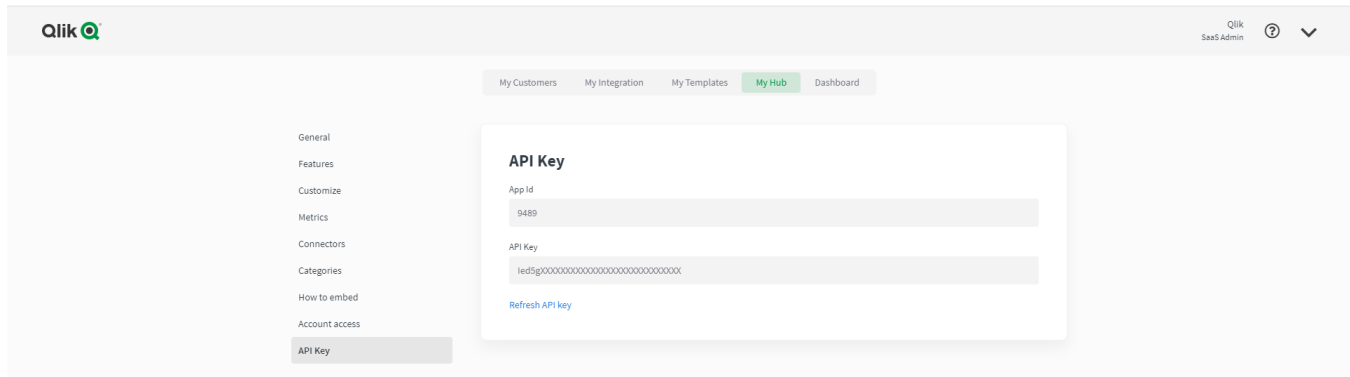
Once SSO is activated, you can download the code from [GitHub](#), unzip the downloaded file, and open the file `server.js` which runs an express server over https.

Scroll down to line 58 to review the JWT authentication flow that generates the signed token.

Only thing you should configure in there for the JWT is the secret and customername, where the secret is your tenant's api key.

How to get your tenant's API key:

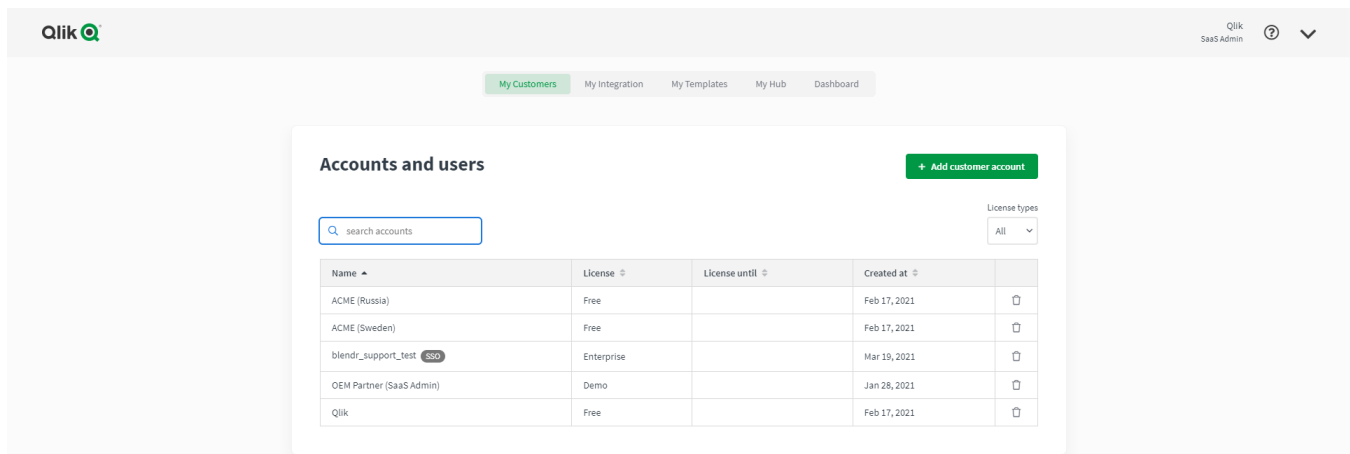
1. Log into your tenant with your SaaS admin account
2. Go to "My Hub -> API Key"
3. API key is masked so you need to create a new one by clicking on "Refresh the API key"
4. Copy the API key and store it in a secured place. This is the secret that we'll use to create the signed JWT token.



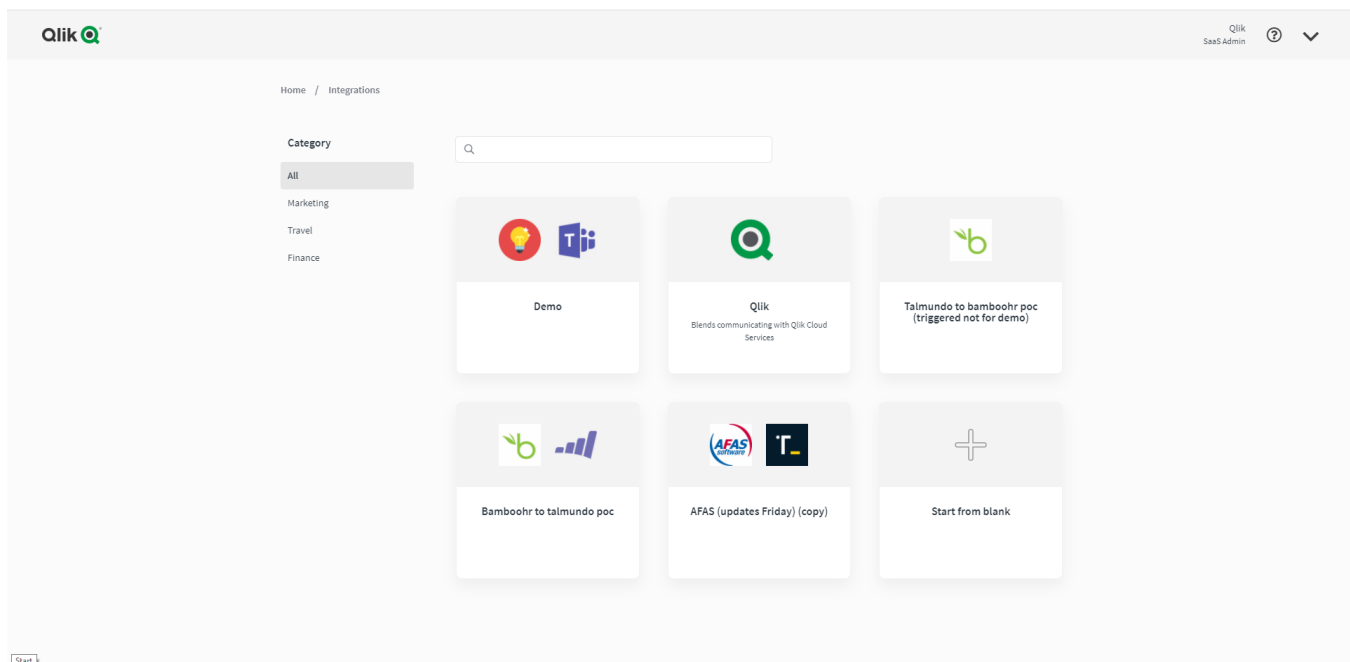
JWT authentication flow:

These are the steps involved to generate the signed token that will be sent as query parameter in the URL:

- Set the *secret* variable in the code using your tenant's API key (line 62)
- Set the *customername* variable with your tenant's name, *alvaropalacios* in my case (line 63)
- The header defines the type of token and the signing algorithm. Use HS256 as the signing algorithm (lines 68-71)
- Then the header is base64url encoded to form the first part of the JWT token (lines 74-75)
- Next is the payload. The *data* variable has a set of predefined claims that are mandatory (lines 78-87). Please check this [link](#) to see all the possible SSO parameters for Blendr as "claims" in the JWT payload.
- SSO must be activated for the *account* you set in the payload. In my case, *blendr_support_test* is the only "customer" with SSO activated in my tenant.



- The payload also needs to be base64url encoded to form the second part of the JWT token (119-120)
- The token will consists of both the header and the data (line 123)
- Now the secret (your tenant's API key) is used to create the signature that will be used in turn to sign the token (line 126)
- The signature also needs to be base64url encoded (line 129)
- Putting all together as signed token. The signed token is three base64url encoded strings separated by dots "." (line 131)
- Lastly, the redirect will send the user from the login page to the Blendr marketplace. The url includes the signed token as query parameter (line 133)



Please check this Introduction to JWT page to get familiar with this flow: <https://jwt.io/introduction>

How to run the nodejs server:

1. Download the files from my [GitHub](#) and unzip in a local folder.
2. On the command line, in the root directory of this project, run "npm install" to create the *node_modules* and download all project dependencies from the package.json file.
3. Now you can run the express server over https. On the command line, type "node server.js"
4. The server is now listening on localhost port 1234. Open Chrome and enter <https://127.0.0.1:1234> to display the login page.
5. You can test using any username and password. When you click on the "Submit" button it will redirect you to your tenant's Hub homepage, e.g. <https://alvaropalacios.admin.blendr.io/templates> (My Hub of templates).
6. If you wish to change the destination page, you can modify line 133 of the code to link to the available integrations (blends), direct link to a template, an integration instance, a link that start a template install, etc. (See [Various SSO links](#))

Log in to Qlik


Username

Password

Submit

Lost your password?

New to site? [Create Account](#)

 Customer Portal

©2016 All Rights Reserved. Gentellella Alelal is a Bootstrap 3 template. Privacy and Terms

If you don't see any templates in your Hub, don't forget to turn a blend into a template. A template is a Blend that has a setting "Is template". Once a Blend is turned into a template, it can be made available to customers of the SaaS partner via the Hub (marketplace).

For more detailed information please visit our online help: [How to create integration Templates](#)

Related articles

- [Blendr.io - SSO example code for SaaS companies](#)
- [Blendr.io - Loop & Reduce functionality for Qlik Sense](#)
- [Blendr.io - Job History Dashboard for OEM \(SaaS\) partners](#)
- [Configure Amazon SES with Qlik Alerting](#)
- [Publish Mashup on Kubernetes Cluster](#)