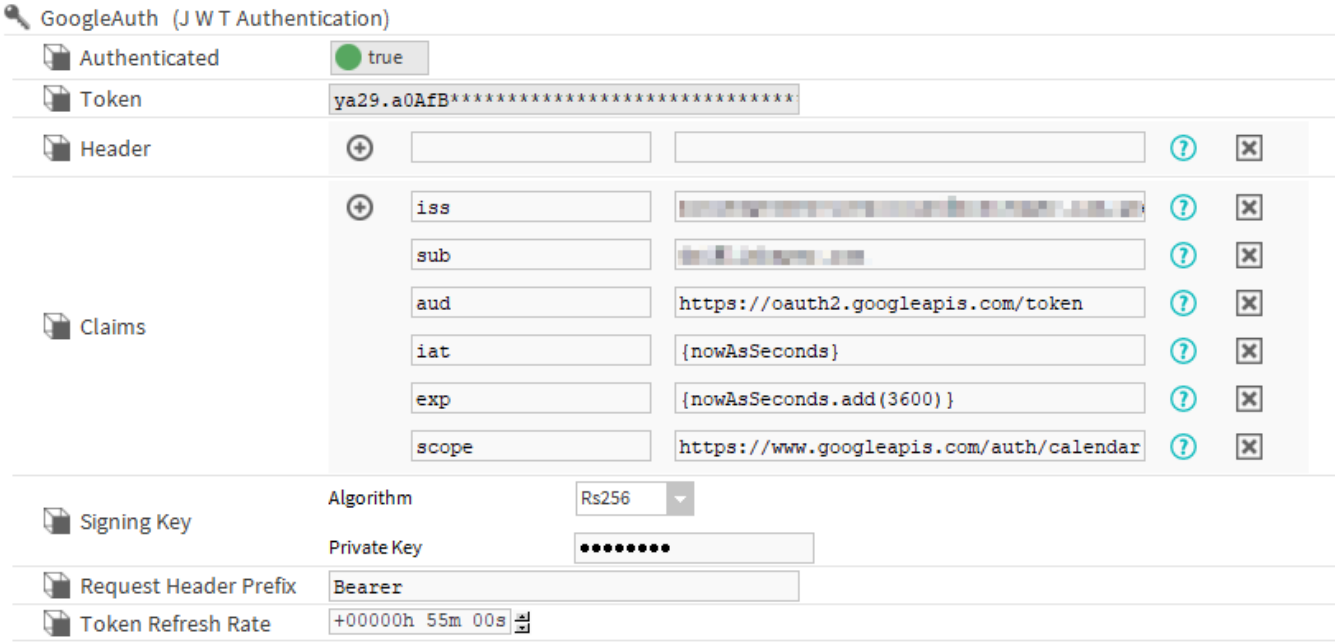
JWTAuth

Summary

The JWTAuth component allows connection to a service by using a JWT.

Implementation

- Drag the authentication from the palette into the AuthenticationContainer of the Http Config.
- Fill the slots
- Trigger the "Refresh Token" action



Properties

- *Authenticated*: Indicates the state of the authentication. If true, a token has been successfully retrieved and this token will be added to the requests.
- *Token*: A glimpse of the token, only the ten first char are displayed
- *Header*: Key Value map allowing you to add values to your header. The "alg" and "typ" value are handled automatically. You can put SFormat in the values (second column)
- *Claims*: Key Value map allowing you to build your Claims. For the "iat" (issuedAt) claim use the {nowAsSeconds} SFormat which retrieves the current time in seconds. For the "exp" (expiration) claim, use the {nowAsSeconds.add(...)} SFormat to get the current time and add your desired amount of seconds. You can put SFormat in the values (second column).
- *SigningKey*: Defines the algorithm used to sign the JWT. The algorithm displays a dropdown of the supported algorithms. Depending on the algorithm you will have different slots,
  - If you selected a HMAC algorithm: you will have the "Secret" slot in which you must add you secret key and the "Secret Base64 Encoded" slot which defines if the key is encoded or not.
  - Otherwise, you will only need to fill the privateKey slot with your private key
- *RequestHeaderPrefix*: The prefix used in the following request to authenticate.
- *TokenRefreshRate*: The interval at which you want to automatically refresh your token. It is advised to set a value slightly lower than the token expiration time.

About the Signing Keys:
If you're using a HMAC algorithm, simply copy paste your key as is. If it is Base64 encoded, set the "Secret Base64 Encoded" slot value to true, and the component will decode it automatically.

If you're using a private key, this component only supports private keys from a PEM String which should look like this (the header and footer may be different):

-----BEGIN PRIVATE KEY-----
MIHgMIGaAgEBMA0GCSqG...
-----END PRIVATE KEY-----

You must copy past the whole string with the header and footer.