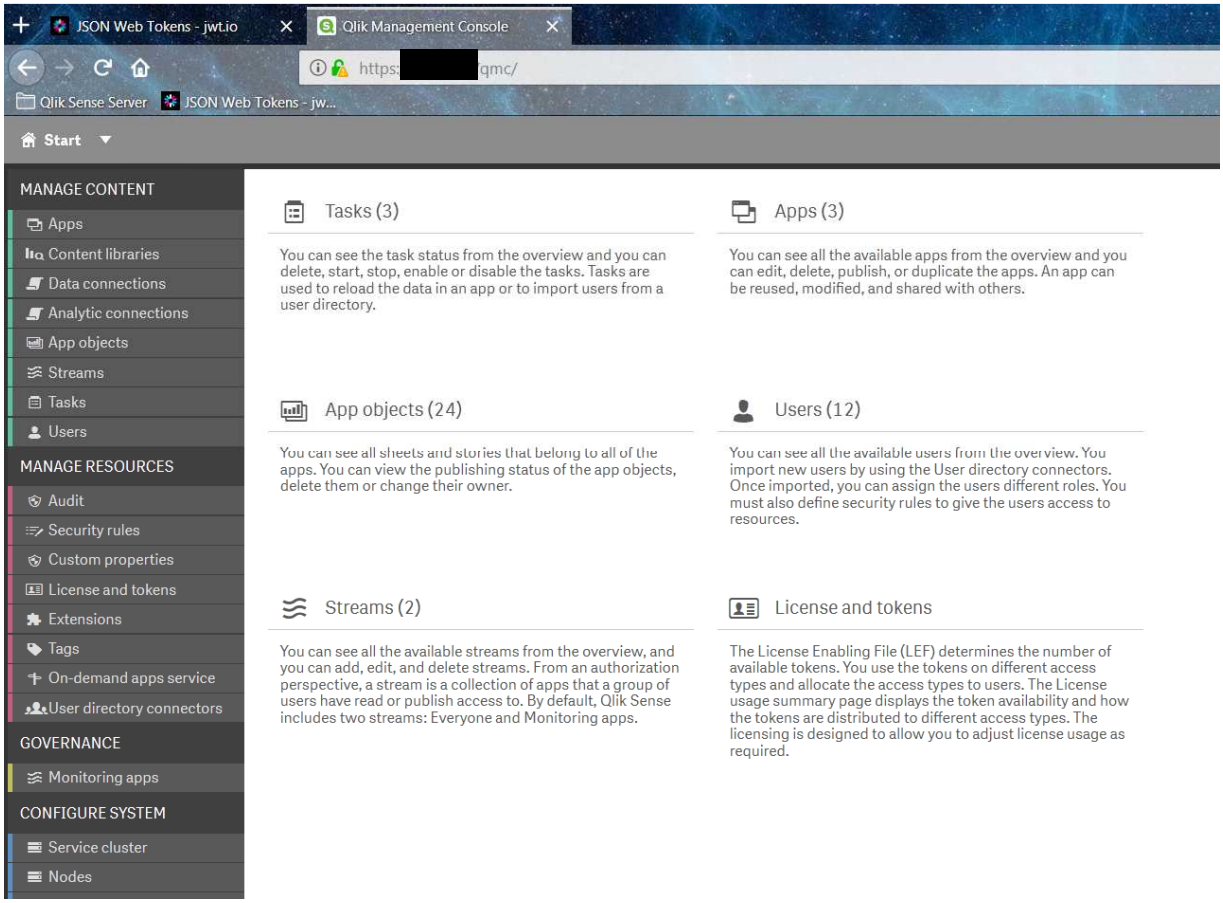
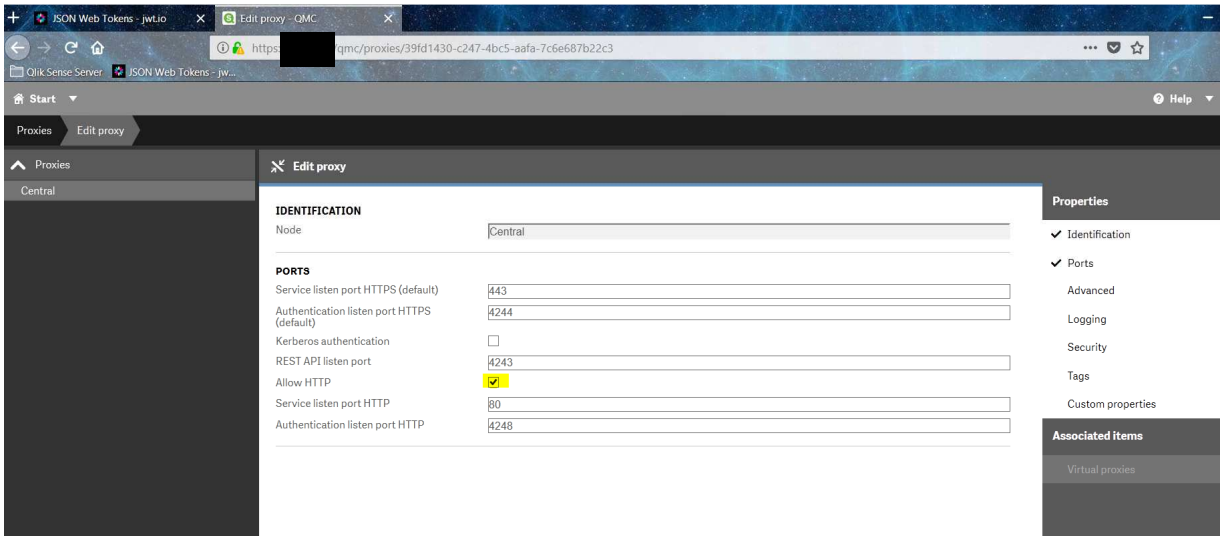
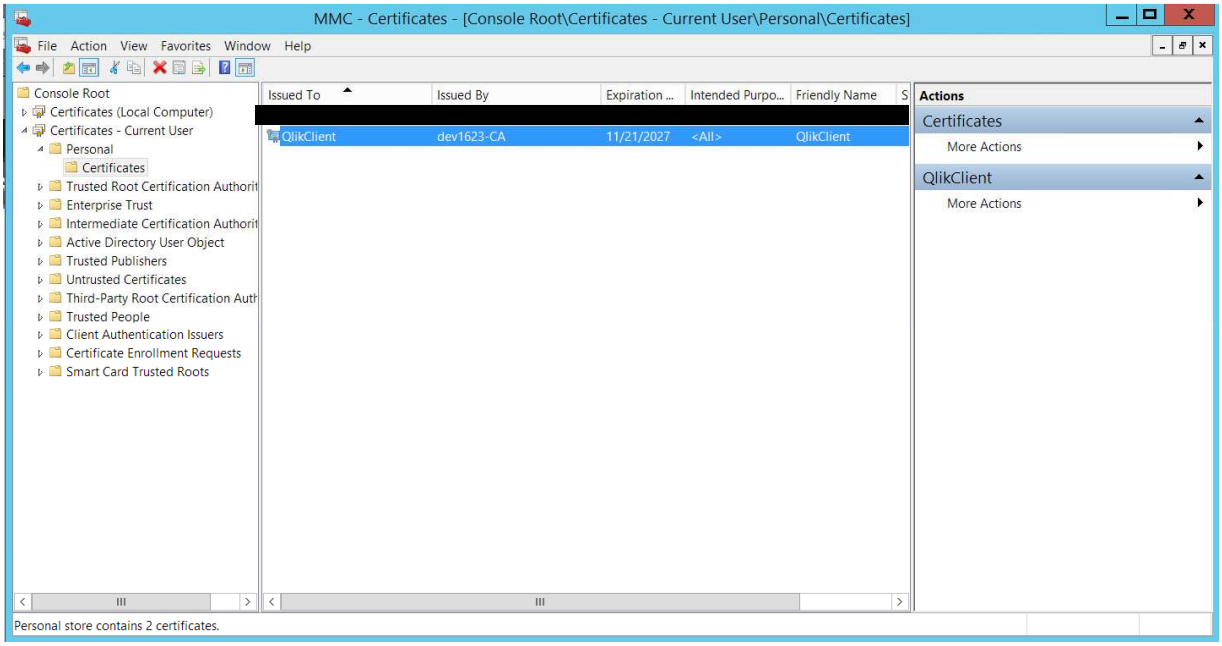
1. Login into QlikSense Management Console



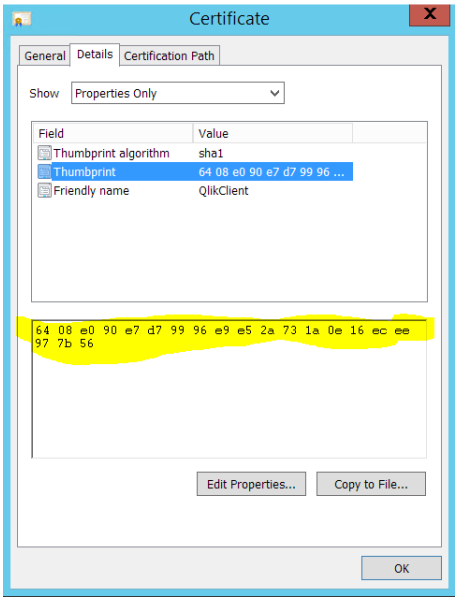
2. Optionally, configure the existing Central proxy to allow for HTTP connections also.



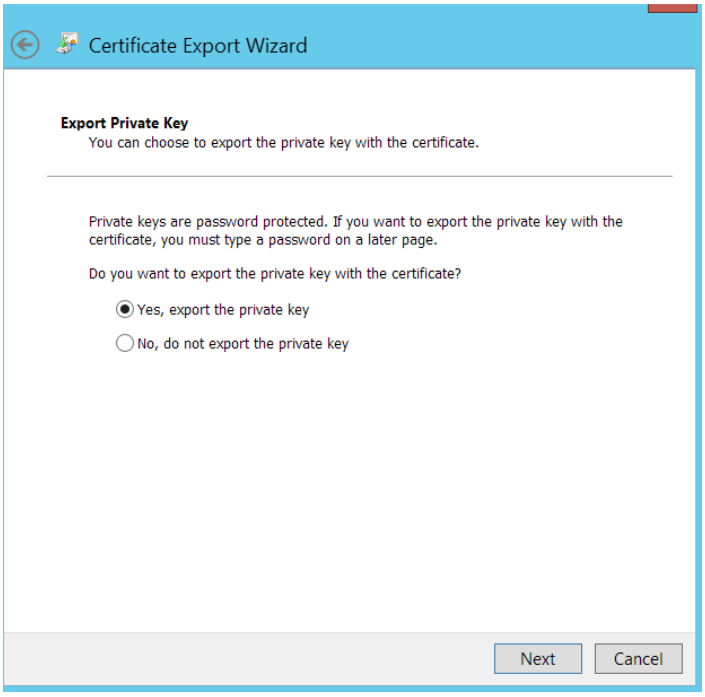
3. Head over the MMC to view the certificates, and add the Certificates snap in for local computer as well as current user.

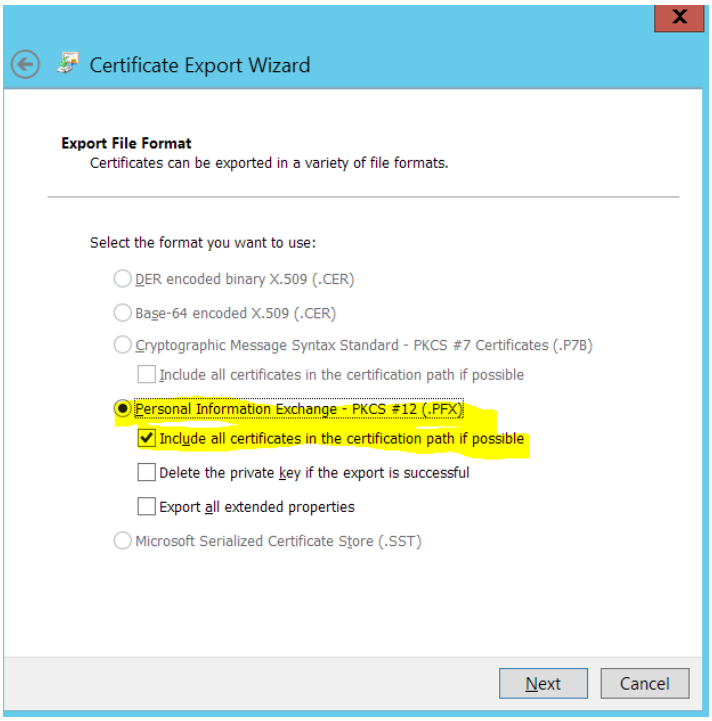


4. Pick the certificate you inserted, or the one Qlik created when starting up. For simplicity, I used the one Qlik created. Open the certificate, and copy the thumbprint text, and save it. It would be used down the line.



5. Export the certificate, with all keys.



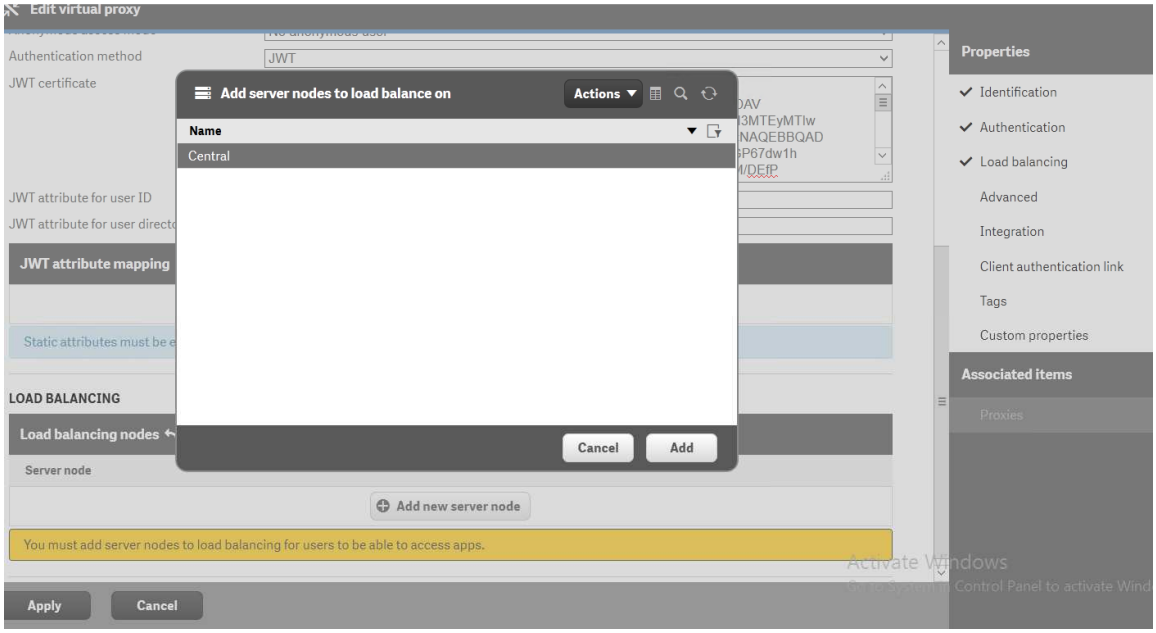


6. Install OpenSSL.

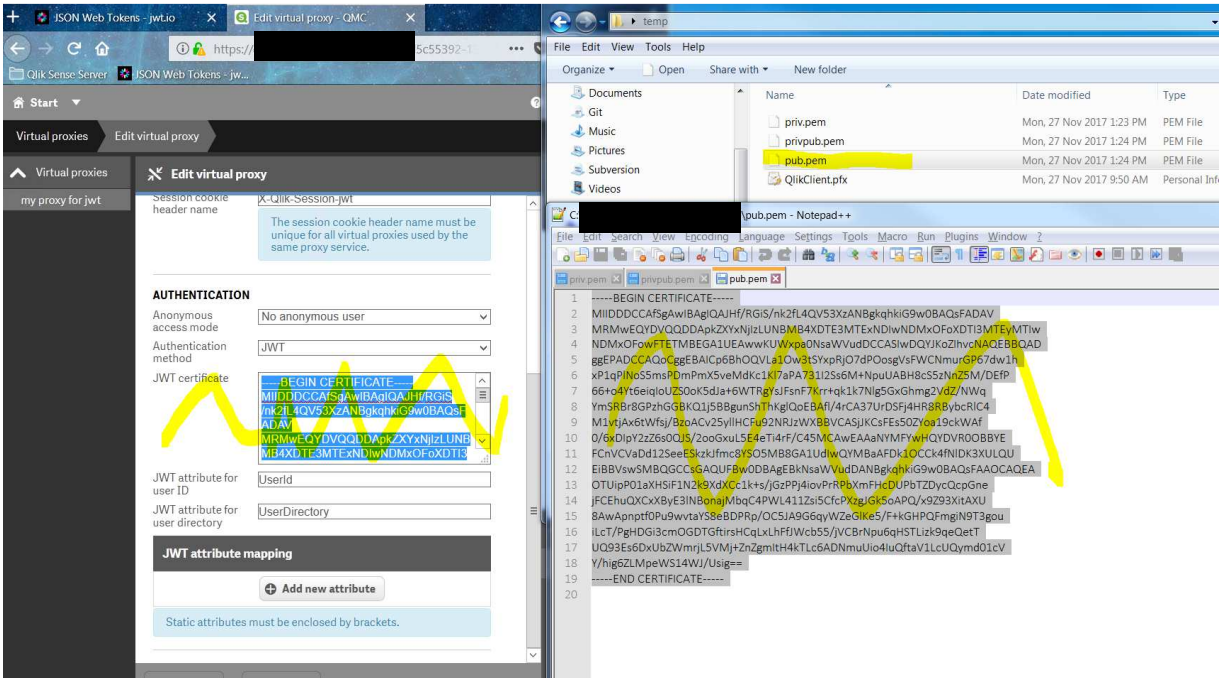
7. Place the exported certificate in a separate folder.
    a. Execute the below commands in a command prompt (in the folder).
        i. openssl.exe pkcs12 -in QlikClient.pfx -nocerts -out priv.pem
        ii. openssl.exe rsa -in priv.pem -out priv.pem
        iii. openssl.exe pkcs12 -in QlikClient.pfx -out privpub.pem
        iv. openssl x509 -inform pem -in privpub.pem -pubkey -out pub.pem -outform pem
    b. Copy the public key to a text file to be used later.

```
C:\Users[redacted]\Desktop\temp
λ ls
QlikClient.pfx

C:\Users\[redacted]Desktop\temp
λ Openssl.exe pkcs12 -in QlikClient.pfx -nocerts -out priv.pem
Enter Import Password:
MAC verified OK
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

C:\Users\[redacted]Desktop\temp
λ Openssl.exe rsa -in priv.pem -out priv.pem
Enter pass phrase for priv.pem:
writing RSA key

C:\Users\[redacted]Desktop\temp
λ openssl.exe pkcs12 -in QlikClient.pfx -out privpub.pem
Enter Import Password:
MAC verified OK
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

C:\Users\[redacted]Desktop\temp
λ openssl x509 -inform pem -in privpub.pem -pubkey -out pub.pem -outform pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAgKnoGE5BUtrU7De1JjGl
GM7t086iyBWwVYI2a6sY/rt3DWHE/Wo+U2hLmaw8OY+Zfm94x0pzUqXto8DvfUjZ
Kzoz42m5QAEfxxLnM2dnkz8MR8/rr6jhi3p6KoihRlLSgrl0lr7pZNGBiwkWycXs
quv6qTWTs2WDkbEaGaDZV1n81apiZJEGvwY/OEYYEpDWPkEGC6dKFOEqCVCgQEB+
X/isIDftSsNIWPgdHxEHJtxGULgzW+2MDHq1Z+yP8HOgAK/bnKWUcIW73Y1EnNZc
EFUIBKMkoKwUSznRlihrX1yRYB/T/rEMiljbNnqzRAlL/aigbG4vkTh5OLisX8Lj
kwIDAQAB
-----END PUBLIC KEY-----

C:\Users\[redacted]Desktop\temp
λ
```

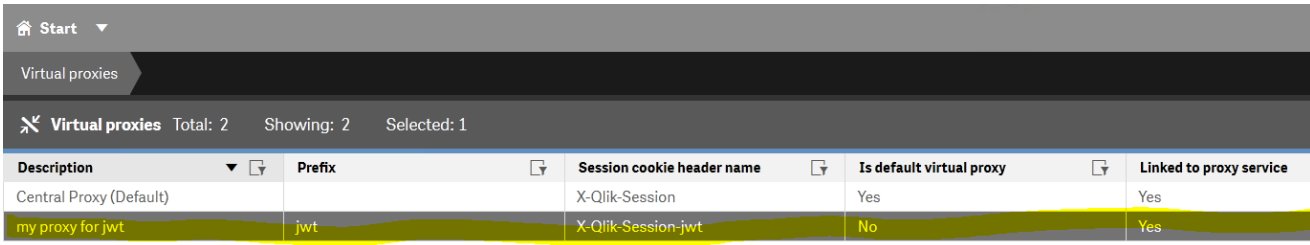
8. Head over to Virtual Proxies screen. Click on “Create New” and fill in the details.
   Be careful about the case, especially when filling in attributes.
   a. Identification
      i. Description: Description to identify the proxy in the virtual proxies screen
      ii. Prefix: This would be the url on which user would access the application
         1. Example: default url: https://<server>/qmc
         2. Proxy prefix: jwt
         3. New url via proxy: https://<server>/jwt/qmc
   b. Authentication
      i. Anonymous access mode: No anonymous user
      ii. Authentication mode: JWT
      iii. JWT certificate: In the folder we used in step 7 above, open the file pub.pem in notepad or similar application, copy the entire text, and paste the same in the text area.
      iv. JWT attribute for user ID: UserId
      v. JWT attribute for user directory: UserDirectory
   c. Click on Load Balancing and add the Server to be used by this proxy.



d. Click Apply and save the new virtual proxy.

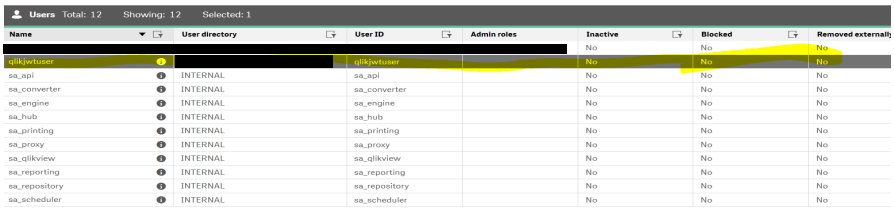


e. By the end of this step, the virtual proxy screen should show our newly created proxy created, and linked to proxy service.



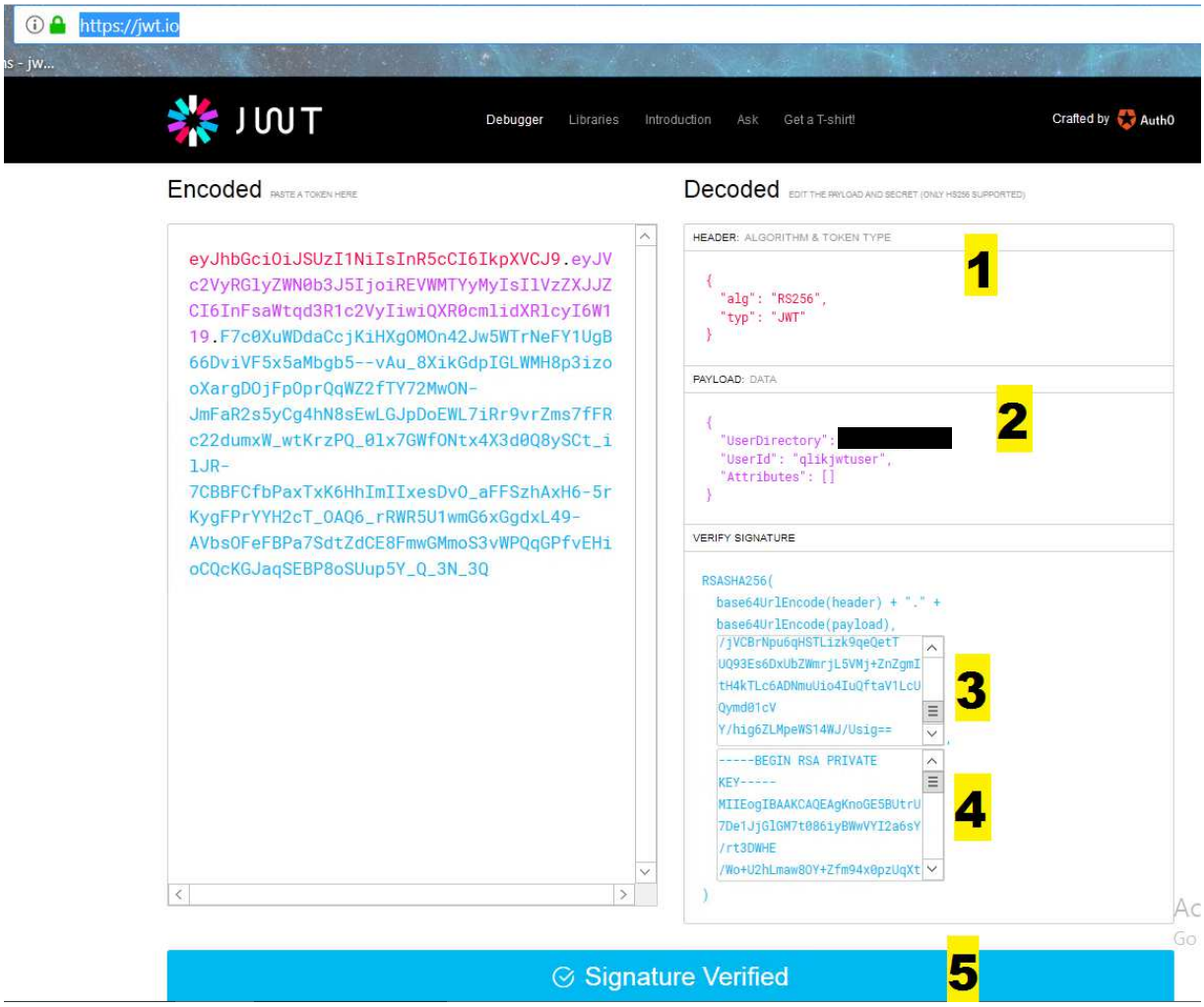
9. Think it's over… no, not so soon ☺ Scroll down for more.

10. Head to the users section in QMC, and create/select the user we wish to use with the proxy we created to test if it works as intended.
    a. Make a note of the user id, user directory, and any assigned roles.



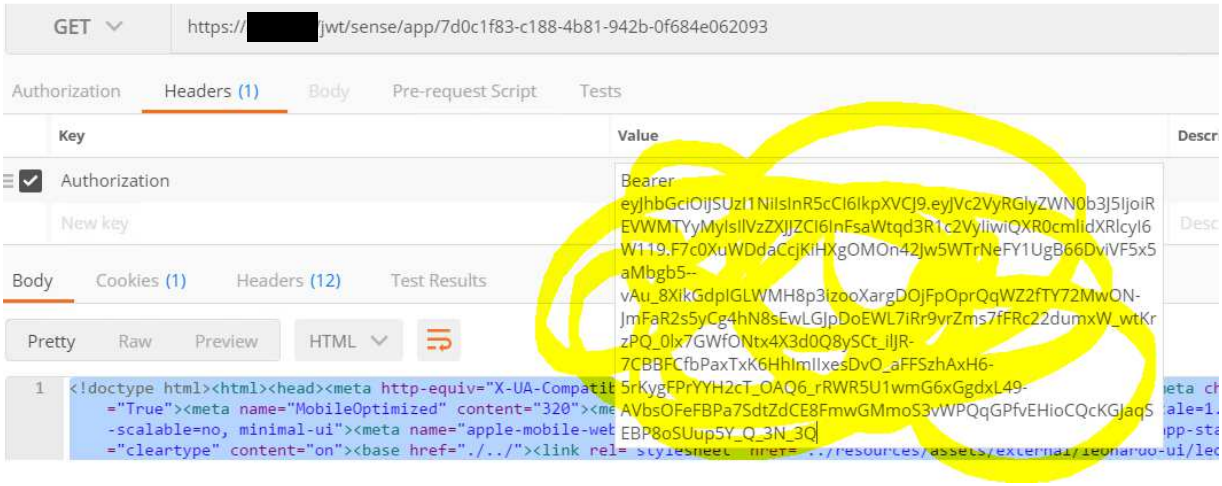
Name	User directory	User ID	Admin roles	Inactive	Blocked	Removed externally
[redacted]				No	No	No
qlikjwtuser	[redacted]	qlikjwtuser		No	No	No
sa_api	INTERNAL	sa_api		No	No	No
sa_converter	INTERNAL	sa_converter		No	No	No
sa_engine	INTERNAL	sa_engine		No	No	No
sa_hub	INTERNAL	sa_hub		No	No	No
sa_printing	INTERNAL	sa_printing		No	No	No
sa_proxy	INTERNAL	sa_proxy		No	No	No
sa_qlikview	INTERNAL	sa_qlikview		No	No	No
sa_reporting	INTERNAL	sa_reporting		No	No	No
sa_repository	INTERNAL	sa_repository		No	No	No
sa_scheduler	INTERNAL	sa_scheduler		No	No	No

11. Open browser window, and open the site https://jwt.io
    a. Segment 1: leave as is
    b. Segment 2: Update userid, user directory, and any attributes of the user (optional)
    c. Segment 3: Paste the public key or the certificate contents
    d. Segment 4: Paste the private key (this is only for testing the generated key)
    e. Segment 5: Make sure that this says “Signature verified”. If this shows otherwise, make necessary changes in the above segments.

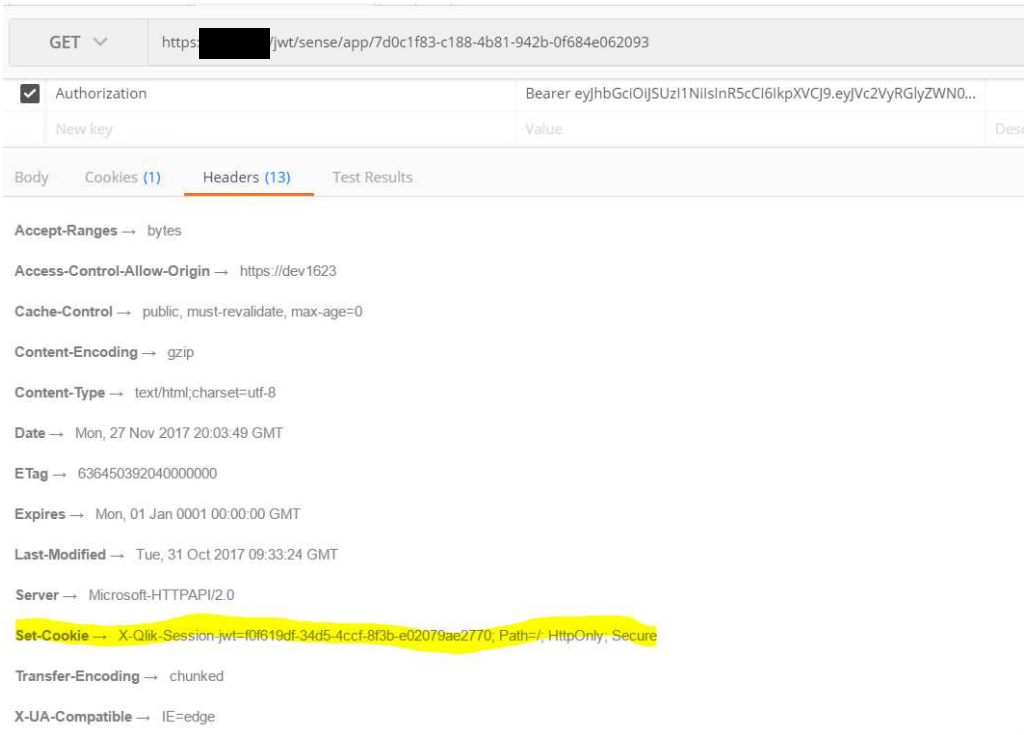


12. Install any tool to place requests to Qlik Sense server. I am using Postman.

13. Open postman (or any tool that supports placing requests).
    a. Set the request to GET
    b. Paste the server url (ensure you have /jwt between server and target page).
       https://<server>**/jwt**/sense/app/<app id>
    c. Click on Headers
        i. Add a new header – Authorization
        ii. Under the value enter as below
            Bearer <space> <encoded value from step 11 above>



14. To ensure it is all working fine, verify the following:
    a. The response should not show a error 401, and/or a qlik page saying authentication failed at proxy.
    b. Ensure the response headers show valid values, and the body portion has valid HTML/data.


GET
https[redacted]/jwt/sense/app/7d0c1f83-c188-4b81-942b-0f684e062093
Authorization
Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJVc2VyRGlyZWN0...
New key
Value
Body
Cookies (1)
Headers (13)
Test Results
Accept-Ranges → bytes
Access-Control-Allow-Origin → https://dev1623
Cache-Control → public, must-revalidate, max-age=0
Content-Encoding → gzip
Content-Type → text/html;charset=utf-8
Date → Mon, 27 Nov 2017 20:03:49 GMT
ETag → 636450392040000000
Expires → Mon, 01 Jan 0001 00:00:00 GMT
Last-Modified → Tue, 31 Oct 2017 09:33:24 GMT
Server → Microsoft-HTTPAPI/2.0
Set-Cookie → X-Qlik-Session-jwt=f0f619df-34d5-4ccf-8f3b-e02079ae2770; Path=/; HttpOnly; Secure
Transfer-Encoding → chunked
X-UA-Compatible → IE=edge