**Disable REST HTTP Server Adapters**

The BFGUI uses its own versions of the REST HTTP Server Adapter. The out-of-the-box version needs to be disabled.

1. From the Administration Menu, select Deployment > Services > Configuration
2. Search for “REST Http Server Adapter”
3. Disable the adapter as it’s a default adapter.
4. Repeat for each node

**Deploy BFGUI\_v1.0-0**

This step deploys the bfgui and sfgerror properties to the database, consolidates any properties on disk, creates the necessary users and permissions and sets up the HTTP Server Adapters.

1. SSH onto SBI node 1
2. Copy the BFGUI v1.0-0.7z deployment package to /opt/ibm/media/bundle
3. cd to the bundle directory created previously (e.g. /opt/ibm/media/bundle)
4. cd to SIR\_BGUI\_v1.0-0/IBM/sbi/scripts/
5. execute 'chmod +x ibm\_deploy\_BFGUI-1.0-0.sh'
6. execute the deployment script './ibm\_deploy\_BFGUI-1.0-0.sh'
7. Accept or update the default bundle home directory (default is /opt/ibm/media/bundle)
8. Accept or update the default SBI install directory (default is /opt/ibm/sbi/install)
9. Node1 only, Enter the username and password for a SBI admin user (required to invoke REST APIs for custom property deployment)
10. The script will now run for some minutes.
11. Repeat all steps for subsequent SBI nodes.

**Edit bfgui.properties for environment**

The environment specific settings in bfgui.properties must be updated (not supported in the environment config script)

1. From the Administration Menu, select Customization > Customization
2. Click ‘Click here to access’
3. Enter your SBI credentials and click Login
4. Click ‘PropertyFile’ on the top menu bar
5. Click ‘bfgui’
6. Click the ‘Property’ tab
7. Search for and update if necessary with the required values, each of the following
   * ***entity.service.swift-inbound***
   * ***entity.swift.service.default***
   * ***link.f5***

**Change BFGAPI User Account password:**

1. From the Administration Menu, select Accounts > User Accounts.
2. Search for “BFGAPIUser”
3. Click ‘edit’
4. Enter and confirm a new password
5. Click ‘Save’
6. Enter the approvers credentials
7. Enter your password
8. Click ‘Finish’

**Copy HTTP Server Adapters for nodes 3 & 4**

The BFGUI deployment script only creates HTTP Server Adapter configurations for node1 and node2. Adapters for node3 and node4 must be created.

1. From the Administration Menu, select Deployment > Services > Configuration
2. Search for ‘FB\_BFGUI\_REST\_SPRT\_HTTP\_SVR\_ADPT\_N1’
3. Click ‘copy’
4. Change the name to ‘FB\_BFGUI\_REST\_SPRT\_HTTP\_SVR\_ADPT\_N3’
5. Set the Environment as ‘node3’
6. Click ‘next’
7. Set the Perimeter Server Name as ‘node3 & local’
8. Click ‘Save’ and ‘Finish
9. From the Administration Menu, select Deployment > Services > Configuration
10. Search for ‘FB\_BFGUI\_REST\_SPRT\_HTTP\_SVR\_ADPT\_N1’
11. Click ‘copy’
12. Change the name to ‘FB\_BFGUI\_REST\_SPRT\_HTTP\_SVR\_ADPT\_N4’
13. Set the Environment as ‘node4’
14. Click ‘next’
15. Set the Perimeter Server Name as ‘node4 & local’
16. Click ‘Save’ and ‘Finish’
17. From the Administration Menu, select Deployment > Services > Configuration
18. Search for ‘FB\_REST\_AUTH\_HTTP\_SVR\_ADPT\_N1’
19. Click ‘copy’
20. Change the name to ‘FB\_REST\_AUTH\_HTTP\_SVR\_ADPT\_N3’
21. Set the Environment as ‘node3’
22. Click ‘next’
23. Set the Perimeter Server Name as ‘node3 & local’
24. Click ‘Save’ and ‘Finish’
25. From the Administration Menu, select Deployment > Services > Configuration
26. Search for ‘FB\_REST\_AUTH\_HTTP\_SVR\_ADPT\_N1’
27. Click ‘copy’
28. Change the name to ‘FB\_REST\_AUTH\_HTTP\_SVR\_ADPT\_N4’
29. Set the Environment as ‘node4’
30. Click ‘next’
31. Set the Perimeter Server Name as ‘node4 & local’
32. Click ‘Save’ and ‘Finish’

**Run environment-config script**

1. Run ***ibm\_rbs\_sir\_applyEnvironmentConfig.sh*** for the ‘BFGUI’ interface and relevant environment using the most recent master settings config sheet.

**Certificate Creation and Import :**

1. Generate CSR file (The reference in the CN is your reference code received from SNOW)

/opt/nfast/bin/openssl req -out CSRBFGUISIRNFT.csr -new -newkey rsa:2048 -nodes -subj "/CN=86993426/OU=Devices/OU=Proving G1 PKI Service/O=The Royal Bank of Scotland Group" -keyout privateKeyBFGSIRNFT.key

1. Obtain DER certificate from portal using CSR and then convert it into pfx

/opt/nfast/bin/openssl pkcs12 -export -out BFGUI\_SIRNFT\_CERTIFICATE.pfx -inkey BFGUI\_SIRNFT\_CERTIFICATE\_Key.txt -in BFGUI\_SIRNFT\_CERTIFICATE.cer  -passout pass:password

1. Use the above file and import that in Sterling B2Bi System Certificates.
2. Rename the BFGUI\_SIRNFT\_CERTIFICATE.pfx to BFGUI\_SIRNFT\_CERTIFICATE.p12 and place the file in “/opt/ibm/sbi/install/liberty/wlp/usr/servers/SIServer/resources/security”.
3. Take the backup of server.xml file present at “/opt/ibm/sbi/install/liberty/wlp/usr/servers/SIServer” and edit the file.
4. Look for “keyStore” xml tag and replace the existing tag with below data after changing the appropriate values. Save the file and restart Sterling B2Bi (we can perform this at last).

<keyStore id="defaultKeyStore" location="BFGUI\_SIRNFT\_CERTIFICATE.p12" password="password"></keyStore>

1. Concatenate the public cert and private cert into a pem file to upload to Tyk

/opt/nfast/bin/openssl x509 -in CSRBFGUISIRNFT.der -out BFGUISIRNFT.pem

/opt/nfast/bin/openssl rsa -in privateKeyBFGSIRNFT.key -out privateKeyBFGSIRNFT.pem

Use cat to concatenate the above both PEM files.

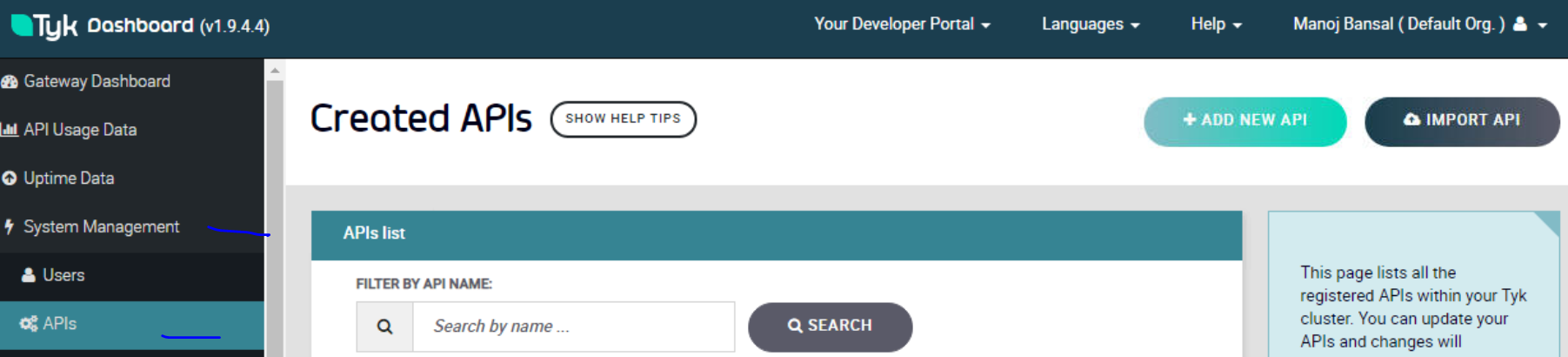
**TYK Configuration :**

Attached are the 3 export files from NFT.

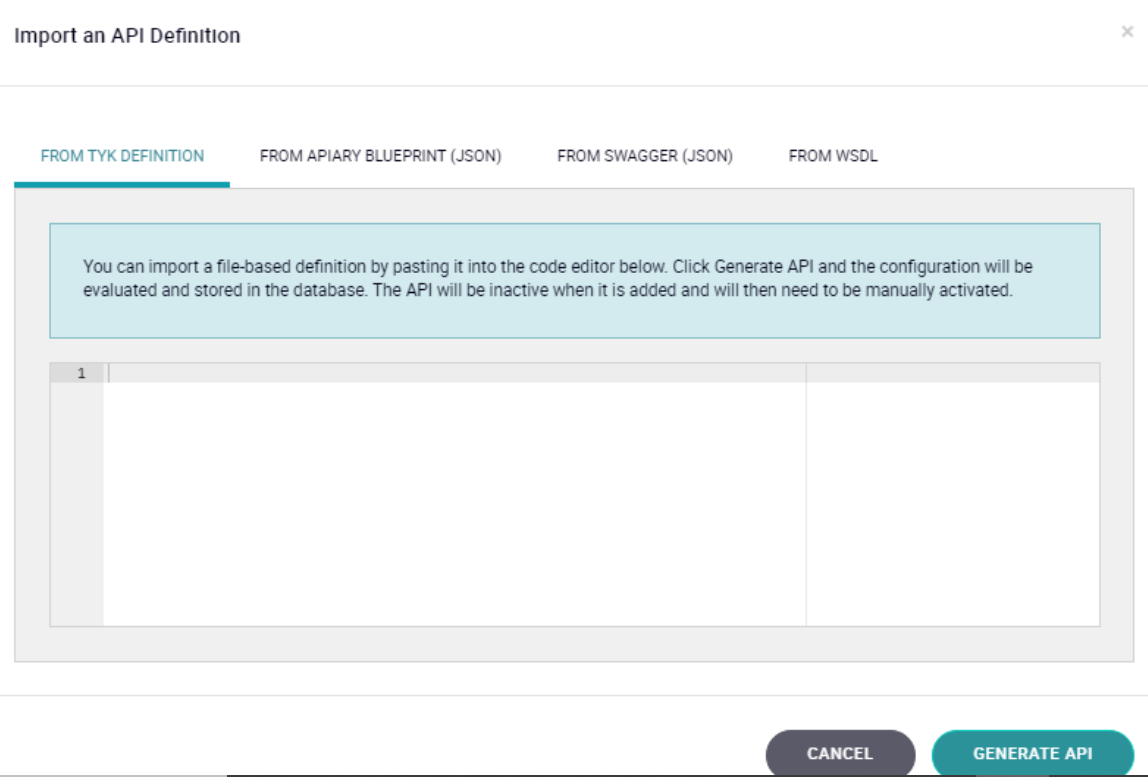


To deploy tyk configuration for PSE or PROD, one need to edit export files (attached above) and update the “name” from “sbi-product-rest-nft” to “sbi-product-rest-<<pse-prod>>” , “target\_list” with the respective environment hostname and also update the listen\_path from "/payments-bfg-sbi-nft/product/rest" to "/payments-bfg-sbi-<<PSE-PROD>>/product/rest".

After updating the json files, open tyk and import the files.

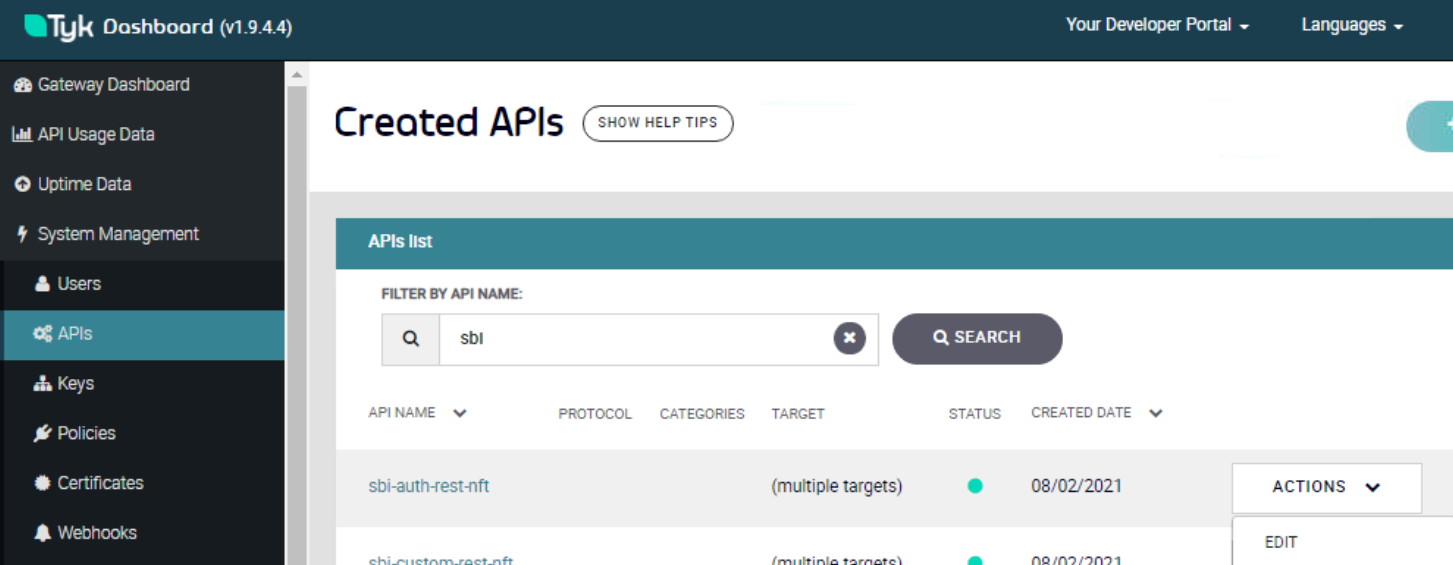


Click on “IMPORT API” and copy paste the content of above json files in the below window

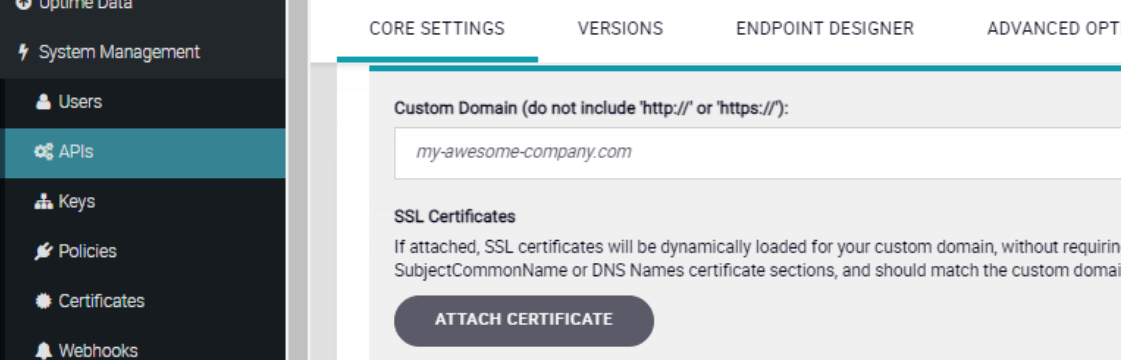


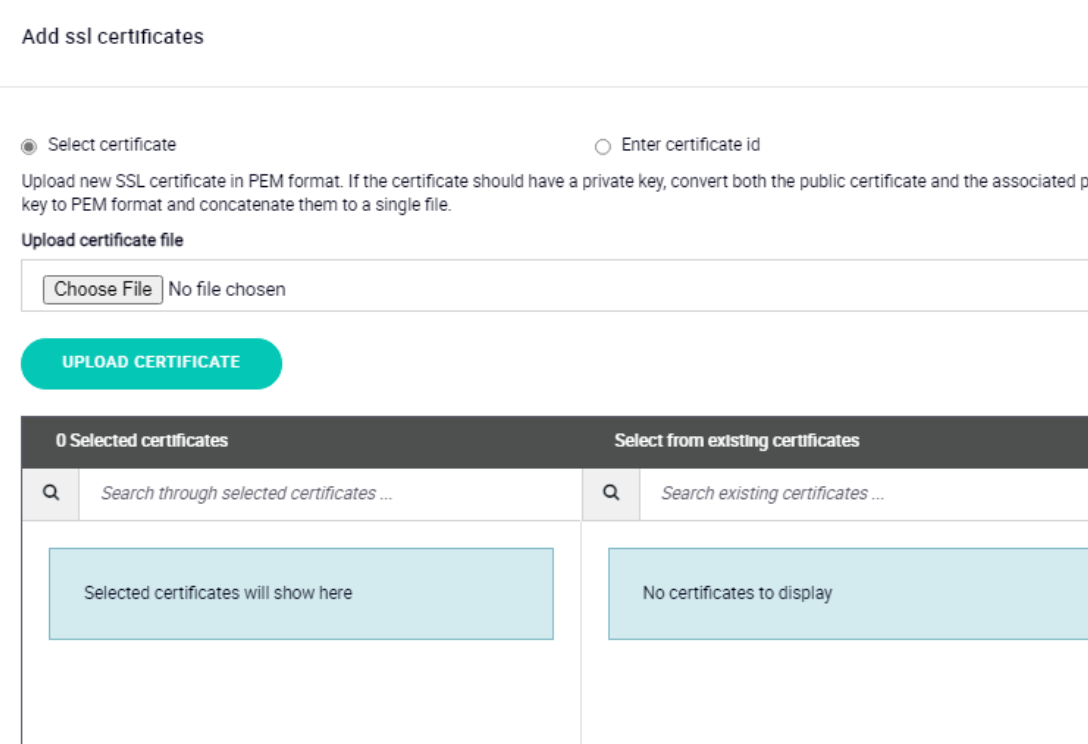
and click on “GENERATE API”.

Now search for 1 of the API and edit it

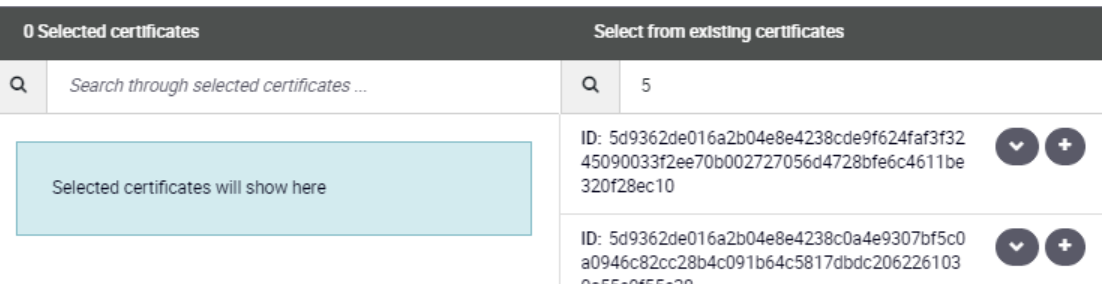


Click on “ATTACH CERTIFICATE”

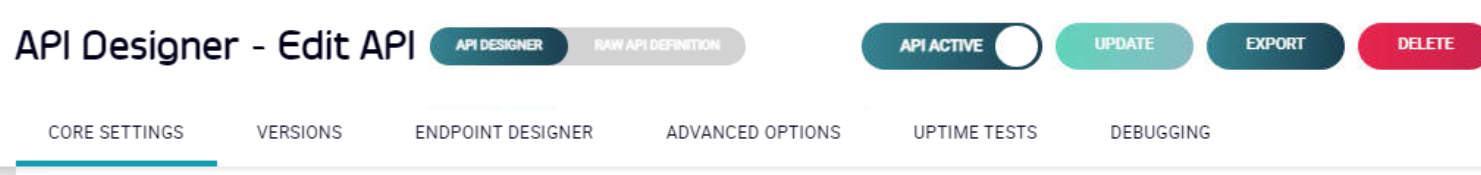


Browse the certificate , upload it and make the note of the certificate id.

Search the certificate with the id and click on Plus(+) sign



Click on “UPDATE” button to save the changes.



Attach the same certificate in other 2 APIs.

**Deploying the BFGUI Application**

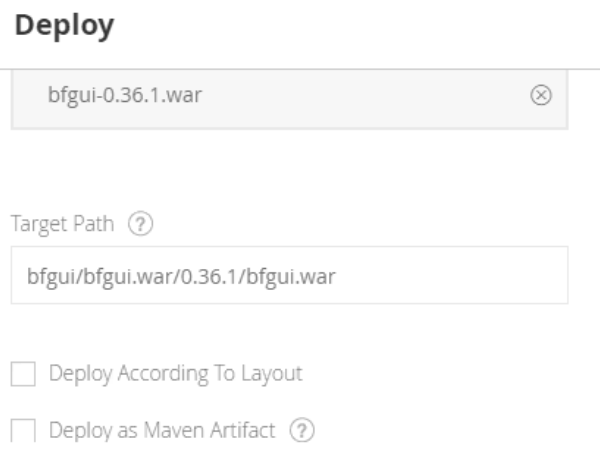
Deploying a new version of the BFGUI has three steps:

1. Upload the new BFGUI WAR version to Artifactory
2. Pull the new version to Urban Code Deploy (UCD)
3. Deploy to the target environment from UCD

Pre-requisites :

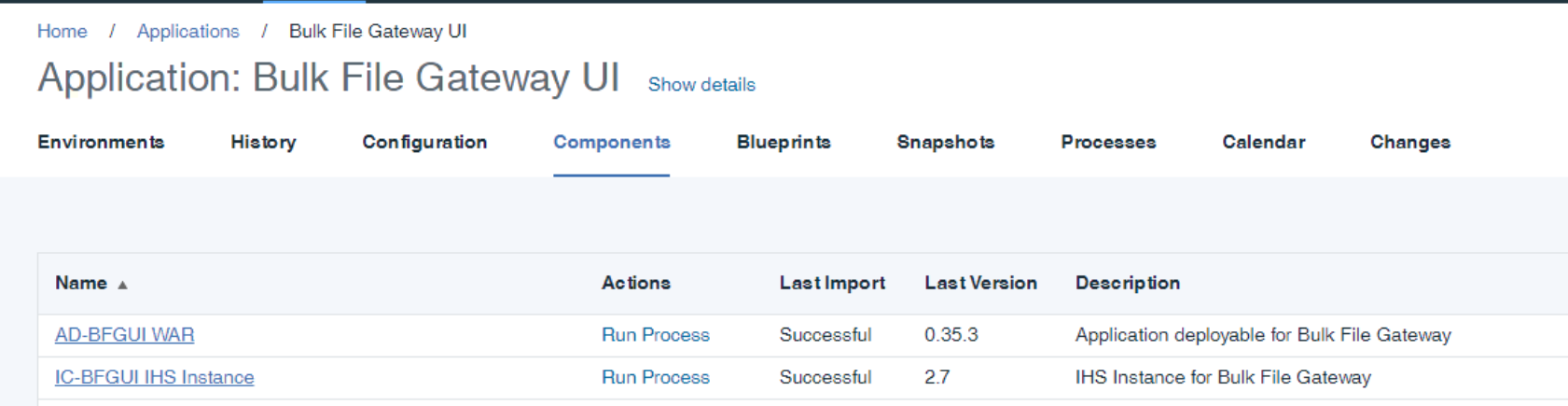
1. you have permissions in the two systems to work with the repositories (Artifactory) and the BFGUI application (UCD)
2. you have locally (file system) a new version of the war, and
3. you know the version number that you want to label the new WAR with e.g. 1.0.2

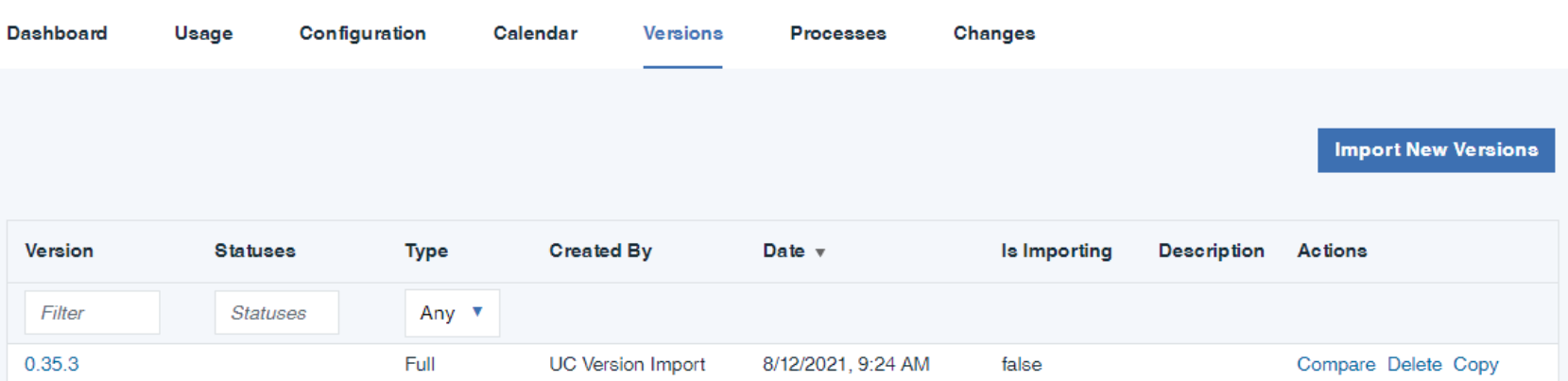
## Upload the new BFGUI WAR version to Artifactory

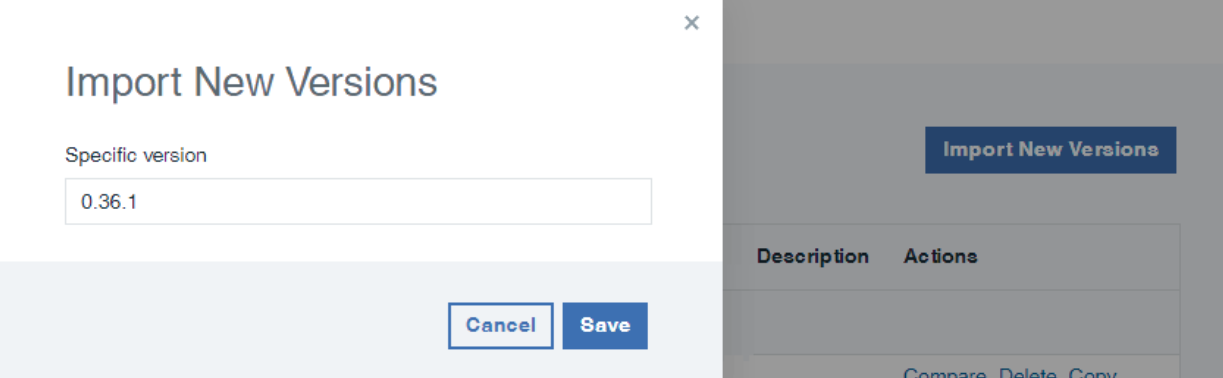
1. Open a browser on Artifactory at the SIR repository root (<https://artifactory.server.rbsgrp.net/artifactory/webapp/#/artifacts/browse/simple/General/sir-maven-all-repos>).
2. Login and select the "Deploy" option on the top right.
3. In the dialog, set the following fields:
   1. Target Repository "sir-maven-ext-local"
   2. "Single file", and browse to the new bfgui war file, e.g. bfgui-0.36.1.war
   3. Uncheck "Deploy as Maven Artifact"
   4. In "Target path", enter (without quotes) "bfgui/bfgui.war/n.n.n/bfgui.war" where n.n.n is the version that you want to upload the war as.  Example bfgui/bfgui.war/0.36.1/bfgui.war.  Make sure the version is in the path and the war file name in the target path is bfgui.war.   The war will be renamed on upload.  
        
      
   5. "Deploy" to upload the war.

## Pull the new version to Urban Code Deploy (UCD)

1. Open a browser on Urban Code Deploy at the BFGUI application (<https://www.ucd-prod.web.rbsgrp.net/#application/1732401b-701f-d392-f4cd-85561b3b7661>)
2. Select "Components" and "AD-BFGUI-WAR":

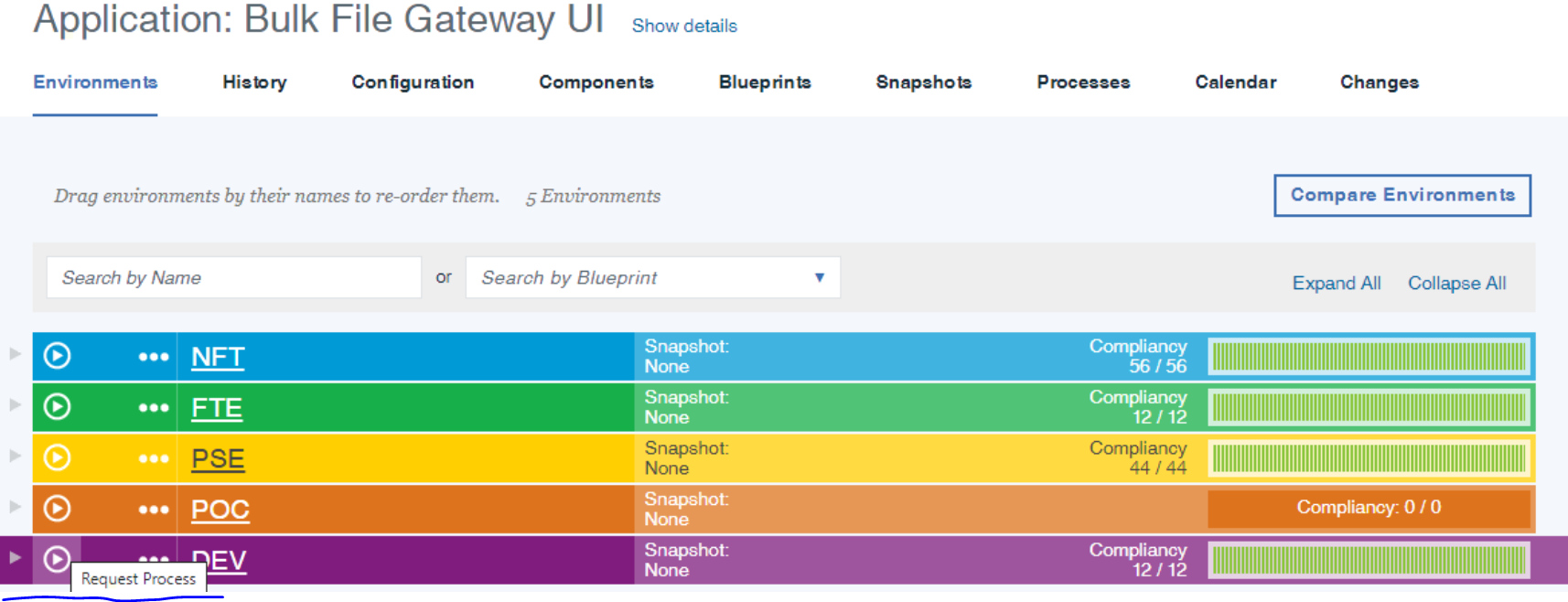
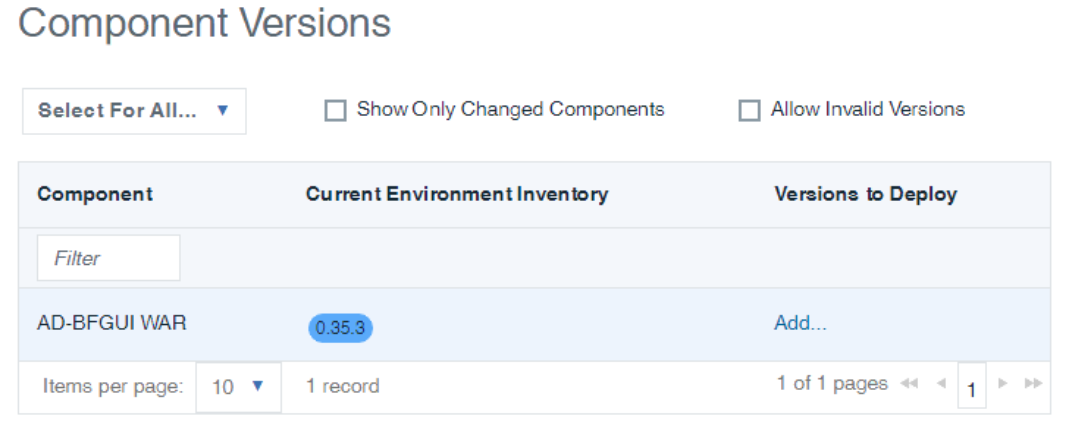


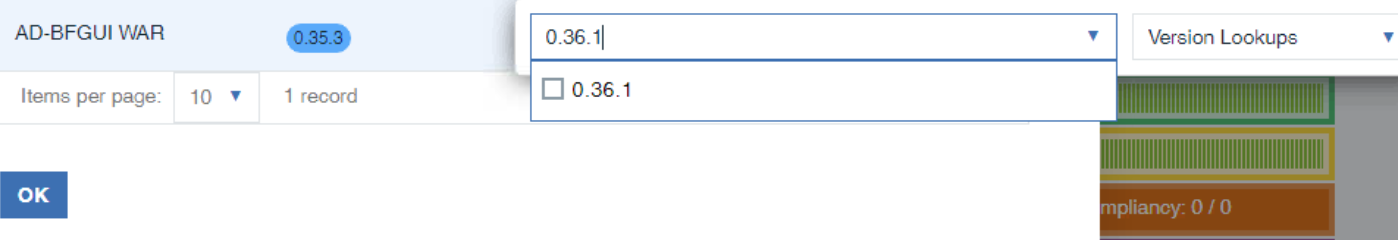
1. In the dialog that appears, you can see the current deployed version.   Select "Versions" from the menu at the top:  
   
2. In the "Versions" screen, on the right, select "Import new versions", In the dialog, enter the version identifier you used in Artifactory eg. 0.36.1 following the example earlier

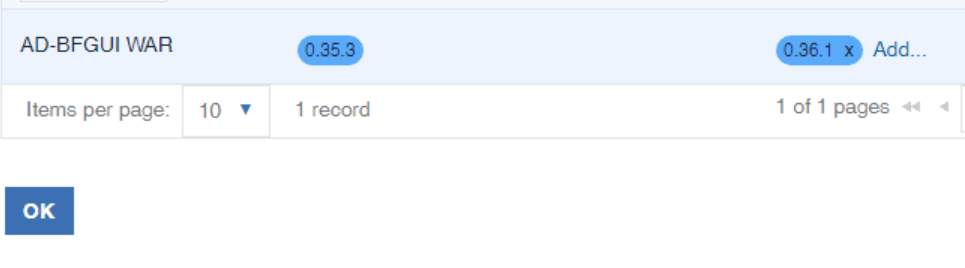


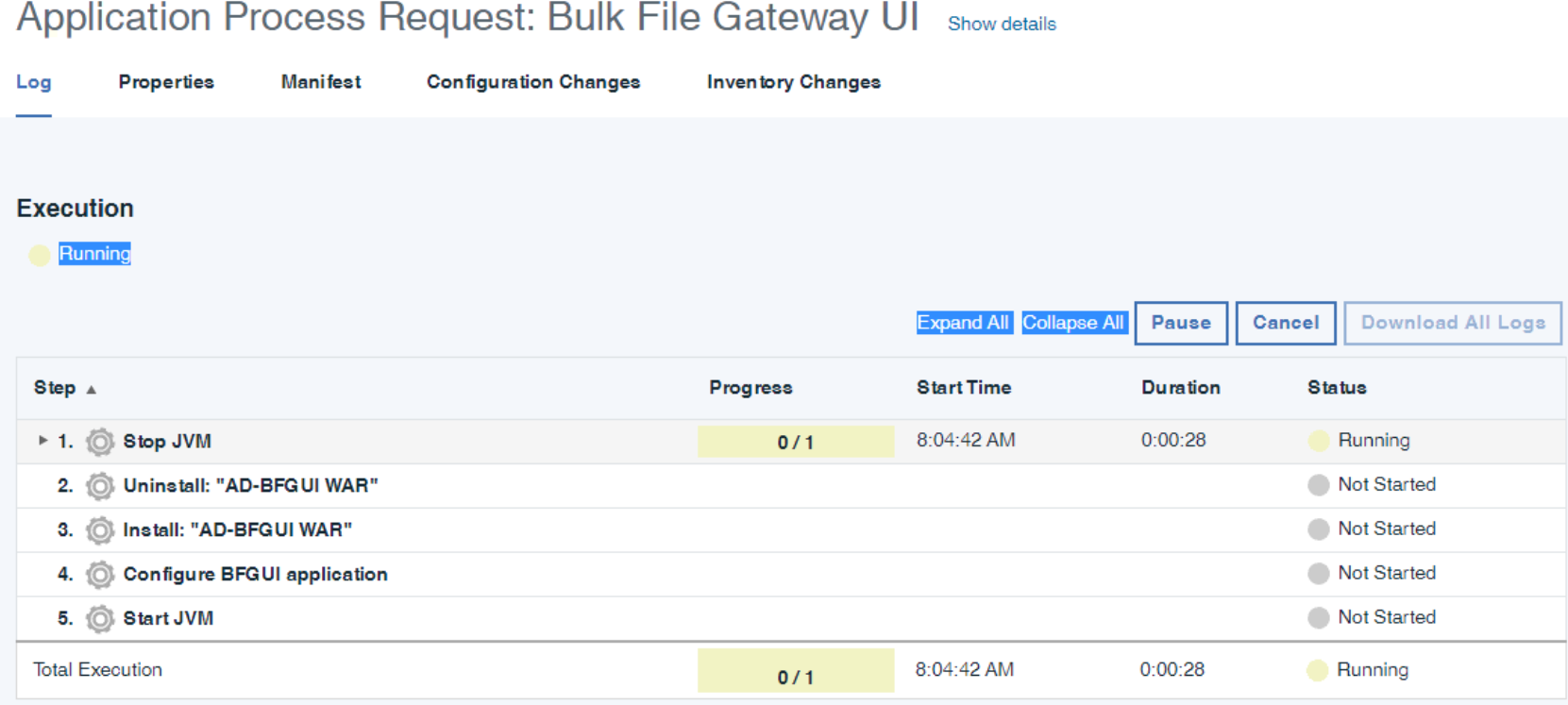
1. Enter "Save" and UCD will import the WAR into the Versions for the BFGUI application.

## Deploy to the target environment from UCD

1. Login (<https://www.ucd-prod.web.rbsgrp.net/#application/1732401b-701f-d392-f4cd-85561b3b7661>) and select "Request process" against the environment to which you want to deploy:  
     
   
2. This will raise a "Run process on {env}" dialog like the following.  Select process "Update BFGUI WAR File":  
   
3. Under component versions, select "Choose version" to raise the following dialog:  
     
   
4. Choose "Add ... " and in the dialog that follows, use the drop down to choose the version identifier you used when uploading/importing the new WAR, and OK to confirm:





1. In the Run process dialog,  with 1 version selected, a description (optional) choose Submit to start the process
2. The process is queued and then executed:  
     
   
3. Successful deployment: - anything else needs troubleshooting:

