SAML SSO in multitenants environment using WSO2 G-Reg 5.4.0 and WSO2 IS 5.10.0

Author	Last updated date	Details
Saranki Magenthirarajah (saranki@wso2.com)	04/03/2021	Added images and labels

Contents

Introduction

This document guides the users on configuring SAML SSO in WSO2 Governance Registry 5.4.0 (WSO2 GReg) and WSO2 Identity Server 5.10.0 (WSO2 IS 5.10.0) to be used in a multitenant environment. The following are the WUM levels of the WSO2 products used for this illustration.

- **WSO2 GReg 5.4.0+1610118916225**
- **WSO2 IS 5.10.0+1613143789209**

The all-in-one pack GReg is used in this scenario. In order to accomplish the SAML SSO requirement the following criterias need to be met.

1. Share the user store between WSO2 GReg and WSO2 IS
2. Use WSO2 IS as the Identity Provider (IdP)

The WSO2 Greg 5.4.0 is a matured product. The database scripts located in **<Product-Home>/dbscripts** folder in both GReg and IS have completely different scripts and are not compatible with each other. Therefore, sharing the databases between GReg and IS is not feasible in these two product versions. As a result we cannot use JDBC user store to share the user details between GReg and IS. In order to overcome this issue we can use LDAP as the user store. WSO2 IS 5.10.0 is shipped with an in-built LDAP LDAP. This IS LDAP will be used as the primary user store and shared between GReg and IS.

Configure WSO2 IS LDAP as the primary user store in WSO2 GReg

We can use the default configurations in WSO2 IS. Therefore, no changes need to be made in the IS pack. We need to first port offset the GReg pack in as IS and GReg cannot run on the same port. This can be achieved via the following configuration.

Port offset GReg with value 1 in **<wso2greg-5.4.0>/repository/conf/carbon.xml** file.

```
<Offset>1</Offset>
```

Publisher's HTTPS port: 9443+1 = 9444

Store's HTTPS port: 9443+1 = 9444

Next, we need to configure the LDAP primary user store in the GReg pack. Therefore, please follow the below mentioned steps.

1. Configure LDAP primary user store in **<wso2greg-5.4.0>/repository/conf/user-mgt.xml** [1]

    a. Comment the default **<UserStoreManager class="org.wso2.carbon.user.core.jdbc.JDBCUserStoreManager">** configuration.

    b. Uncomment the existing <**ISUserStoreManager class="org.wso2.carbon.user.core.ldap.ReadWriteLDAPUserStoreManager" >** configuration and modify it as follows.

        - The inbuilt IS's LDAP user store has been used here. Therefore, the ConnectionURL should be set as **ldap://localhost:10389.** This can be identified in **[user_store].connection_url** located in **<IS-Home>/repository/conf/deployment.toml** file and in **<LDAPServerPort>** located in **<IS-Home>/repository/conf/carbon.xml**

```
<UserStoreManager
class="org.wso2.carbon.user.core.ldap.ReadWriteLDAPUserStoreManager">
        <Property
name="TenantManager">org.wso2.carbon.user.core.tenant.CommonHybridLDAPTenantMan
ager</Property>
        <Property name="ConnectionURL">ldap://localhost:10389</Property>
        <Property name="ConnectionName">uid=admin,ou=system</Property>
        <Property name="ConnectionPassword">admin</Property>
        <Property name="AnonymousBind">false</Property>
        <Property name="UserSearchBase">ou=Users,dc=wso2,dc=org</Property>
        <Property name="UserEntryObjectClass">identityPerson</Property>
        <Property name="UserNameAttribute">uid</Property>
        <Property
name="UserNameSearchFilter">(&amp;(objectClass=person)(uid=?))</Property>
        <Property name="UserNameListFilter">(objectClass=person)</Property>
        <Property name="DisplayNameAttribute"/>
        <Property name="ReadGroups">true</Property>
        <Property name="WriteGroups">true</Property>
        <Property name="GroupSearchBase">ou=Groups,dc=wso2,dc=org</Property>
        <Property name="GroupEntryObjectClass">groupOfNames</Property>
        <Property name="GroupNameAttribute">cn</Property>
        <Property
name="GroupNameSearchFilter">(&amp;(objectClass=groupOfNames)(cn=?))</Property>
        <Property name="GroupNameListFilter">(objectClass=groupOfNames)</Property>
```

```
        <Property name="MembershipAttribute">member</Property>
        <Property name="BackLinksEnabled">false</Property>
        <Property name="UsernameJavaRegEx">[a-zA-Z0-9._-|//]{3,30}$</Property>
        <Property name="UsernameJavaScriptRegEx">^[\S]{3,30}$</Property>
        <Property name="UsernameJavaRegExViolationErrorMsg">Username pattern policy
violated</Property>
        <Property name="PasswordJavaRegEx">^[\S]{5,30}$</Property>
        <Property name="PasswordJavaScriptRegEx">^[\S]{5,30}$</Property>
        <Property name="PasswordJavaRegExViolationErrorMsg">Password length should
be within 5 to 30 characters</Property>
        <Property name="RolenameJavaRegEx">[a-zA-Z0-9._-|//]{3,30}$</Property>
        <Property name="RolenameJavaScriptRegEx">^[\S]{3,30}$</Property>
        <Property name="SCIMEnabled">true</Property>
        <Property name="IsBulkImportSupported">true</Property>
        <Property name="EmptyRolesAllowed">true</Property>
        <Property name="PasswordHashMethod">PLAIN_TEXT</Property>
        <Property name="MultiAttributeSeparator">,</Property>
        <Property name="MaxUserNameListLength">100</Property>
        <Property name="MaxRoleNameListLength">100</Property>
        <Property name="kdcEnabled">false</Property>
        <Property name="defaultRealmName">WSO2.ORG</Property>
        <Property name="UserRolesCacheEnabled">true</Property>
        <Property name="ConnectionPoolingEnabled">false</Property>
        <Property name="LDAPConnectionTimeout">5000</Property>
        <Property name="ReadTimeout"/>
        <Property name="RetryAttempts"/>
    </UserStoreManager>
```

c. Please don't remove the following authorization manager configuration.

```
<AuthorizationManager
class="org.wso2.carbon.user.core.authorization.JDBCAuthorizationManager">
        <Property name="AdminRoleManagementPermissions">/permission</Property>
        <Property name="AuthorizationCacheEnabled">true</Property>
        <Property name="GetAllRolesOfUserEnabled">false</Property>
</AuthorizationManager>
```

Enable SAML SSO in WSO2 GReg Publisher and Store

1. Enable the SAML SSO in WSO2 GReg Publisher

In order to enable the SAML SSO in WSO2 GReg publisher portal please follow the below mentioned steps.

Change the following configurations in the **<wso2greg-5.4.0>/repository/deployment/server/jaggeryapps/publisher/config/publisher.json** file.

- Since the public certificates of Greg and IS were not added to each other we have disabled the "responseSigningEnabled" and "assertionSigningEnabled".

```
"authentication": {
     "activeMethod": "sso",
      "methods": {
          "sso": {
              "attributes": {
                  "issuer": "publisher",
                  "identityProviderURL": "https://<Host or IP address of IS server>:9443/samlsso",
                  "responseSigningEnabled": false,
                  "acs": "https://<Host or IP address of Greg publisher node>:9444/publisher/acs",
                  "identityAlias": "wso2carbon",
                  "defaultNameIDPolicy":
"urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified",
                  "useTenantKey": false,
                  "isPassive":false,
                  "validateAssertionValidityPeriod": true,
                  "validateAudienceRestriction": true,
                  "assertionSigningEnabled": false
                          }
                        },
```

2. Enable the SAML SSO in WSO2 GReg Store

In order to enable the SAML SSO in WSO2 GReg store portal please follow the below mentioned steps.

Change the following configurations in the **<wso2greg-5.4.0>/repository/deployment/server/jaggeryapps/store/config/store.json** file.

- Since the public certificates of Greg and IS were not added to each other we have disabled the "responseSigningEnabled" and "assertionSigningEnabled".

```
"authentication": {
      "activeMethod": "sso",
      "methods": {
          "sso": {
            "attributes": {
                  "issuer": "store",
                  "identityProviderURL": "https://<Host or IP address of IS server>:9443/samlsso",
                  "responseSigningEnabled": false,
                  "acs": "https://<Host or IP address of Greg store node>:9444/store/acs",
                  "identityAlias": "wso2carbon",
                  "defaultNameIDPolicy":
"urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified",
                  "useTenantKey": false,
                  "isPassive":false,
                  "validateAssertionValidityPeriod": true,
                  "validateAudienceRestriction": true,
                  "assertionSigningEnabled": false
                        }
                      },
```

Configure WSO2 IS as the Identity Provider

Follow the steps mentioned below to create two service providers (SP) called 'publisher' and 'store' to configure the WSO2 IS node as the IdP. This needs to be done in WSO2 IS's carbon management console. Therefore, log in to the IS carbon management console as the super tenant admin user (Default credentials - username: admin, password: admin).

1. Create Publisher Service Provider

a. In the carbon management console, click **Main → Service Providers → Add**

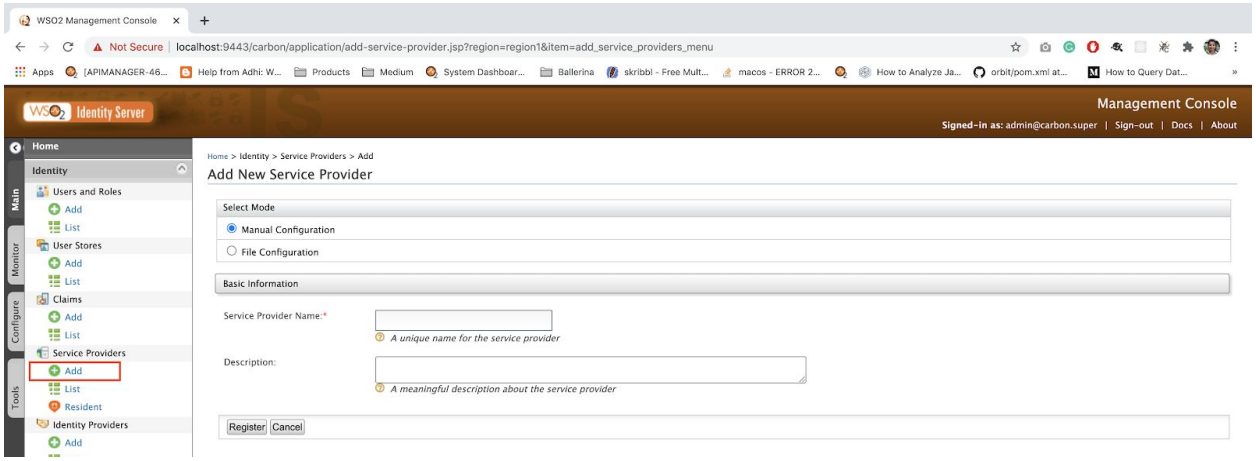

Fig 1: Add a new service provider

b. Enter **'publisher'** for the **Service Provider Name** and click **Register**.

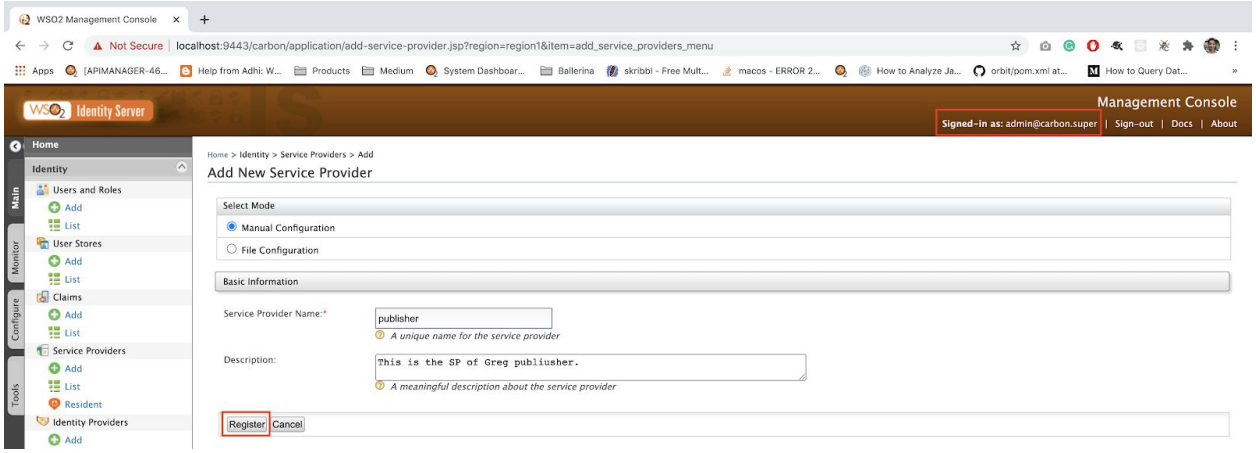

Fig 2: Create publisher SP and register

c. Service Provider details will be displayed and check the "**SaaS Application**" if multi-tenancy is used.

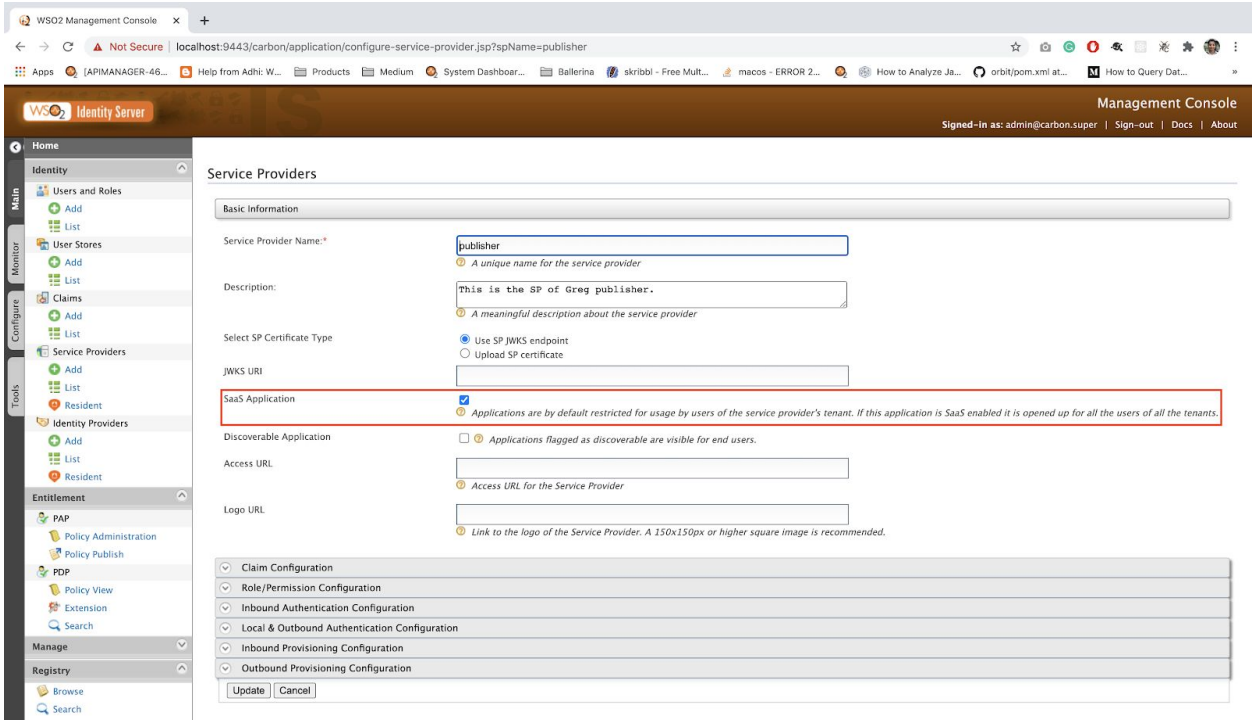

Fig 3: Enable “SaaS Application”

d. Click on **Inbound Authentication Configuration**, next click **SAML2 Web SSO Configuration** and then click **Configure**.

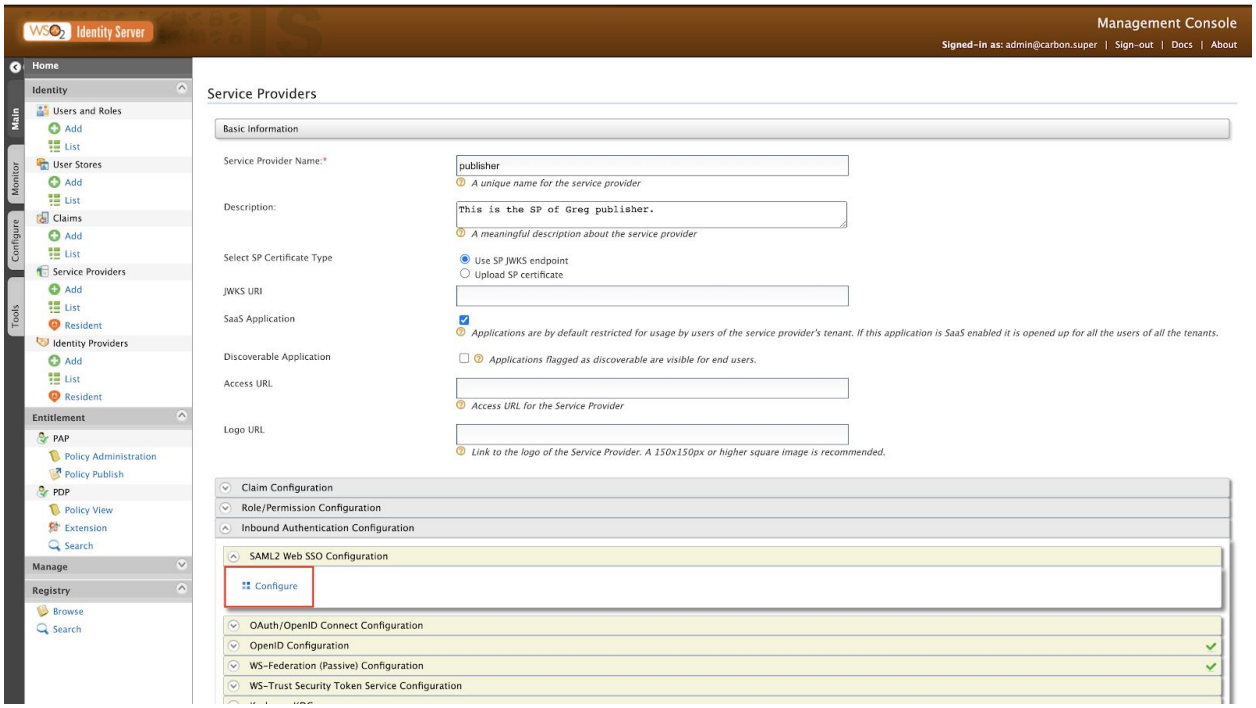

Fig 4: Configure SAML SSO

e. Provide the following details to the created publisher SP.

   i. **Issuer:** Enter 'publisher' for this. Note that the service provider name you provide here should be the same as the issuer value configured in **<wso2greg-5.4.0>/repository/deployment/server/jaggeryapps/publisher/config/publisher.json** file.

   ii. **Assertion Consumer URLs:** Enter the URL of the Publisher node, and click Add.

- Since the public certificates of Greg and IS were not added to each other we have unchecked the "**Enable Response Signing**", "**Enable Signature Validation in Authentication Requests and Logout Requests**", and "**Enable Assertion Encryption**".

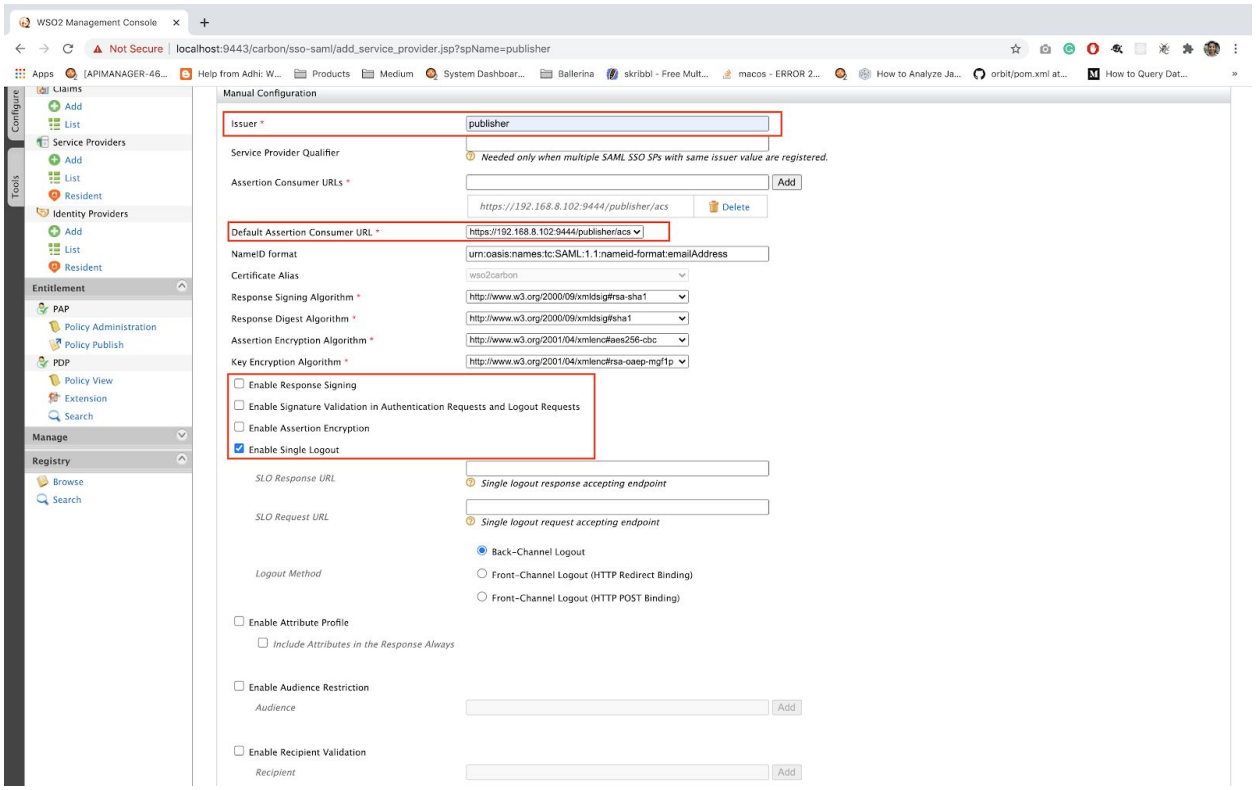

Fig 5: Configure publisher SP

f. Click Register.

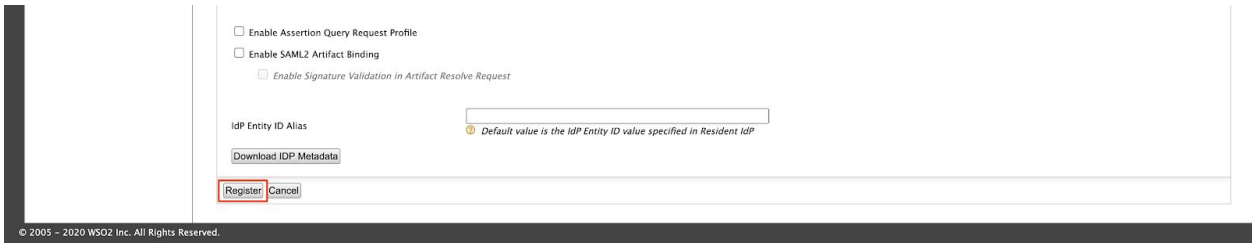

Fig 6: Register the publisher SP

g. Click on **Local & Outbound Authentication Configuration** and check **Use tenant domain in local subject identifier.**

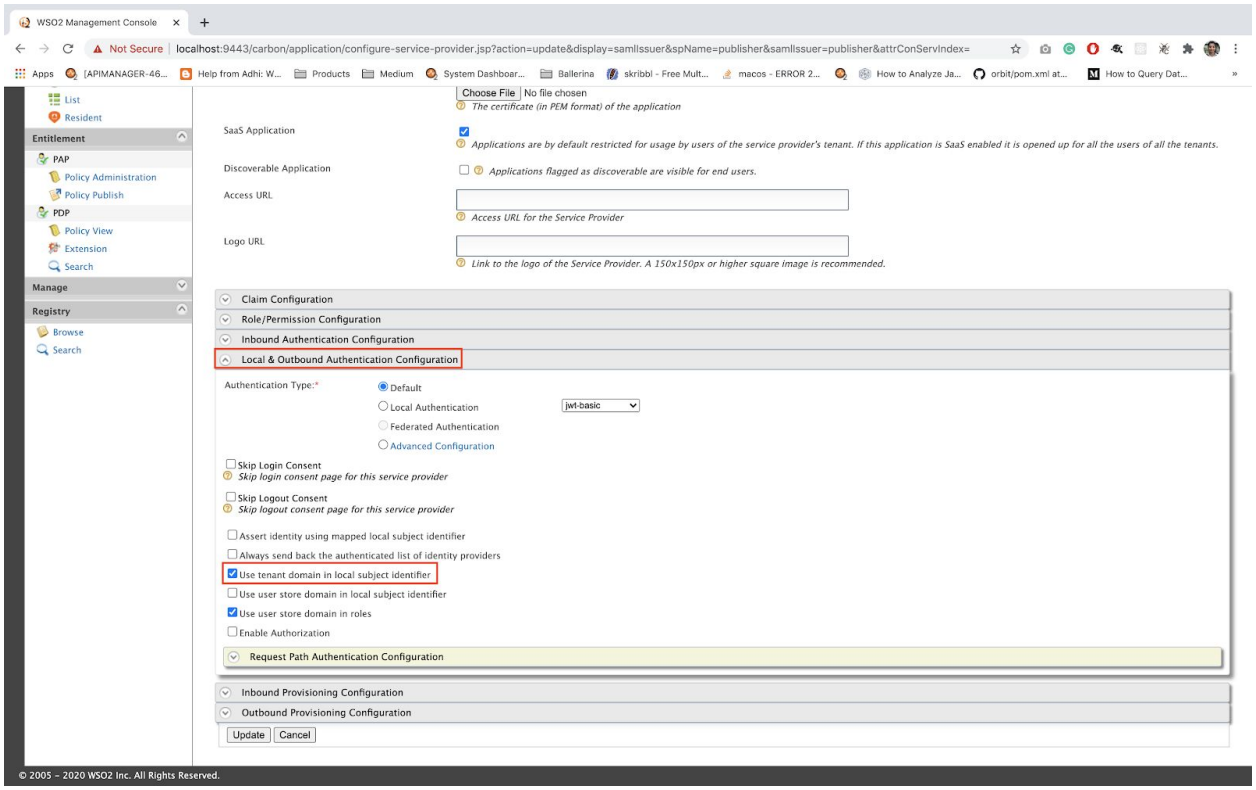

Fig 7: Configure Outbound and Authentication

h. Update the publisher SP configurations.

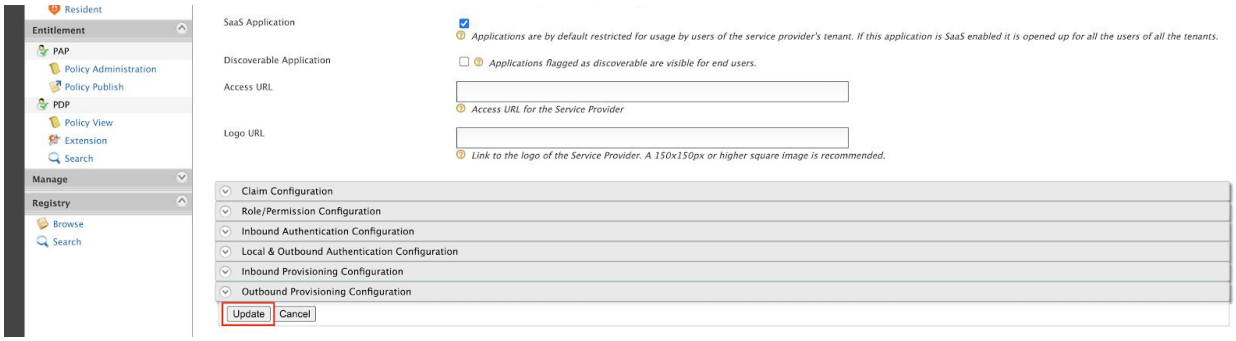

Fig 8: Update the publisher SP configurations

i. The created publisher SP will be listed as below.

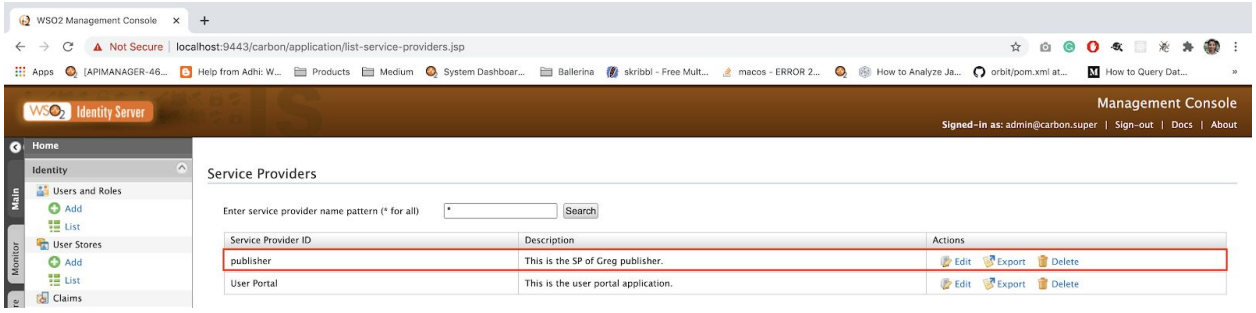

Fig 9: List the publisher SP

2. Create Store Service Provider

a. In the carbon management console, click **Main → Service Providers → Add**

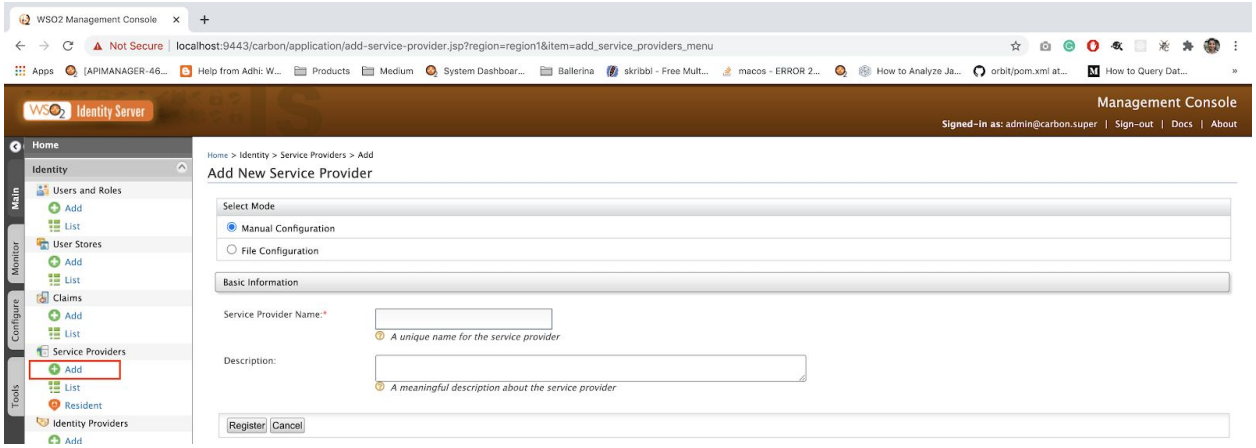

Fig 10: Add a new service provider

b. Enter '**store'** for the **Service Provider Name** and click **Register**.

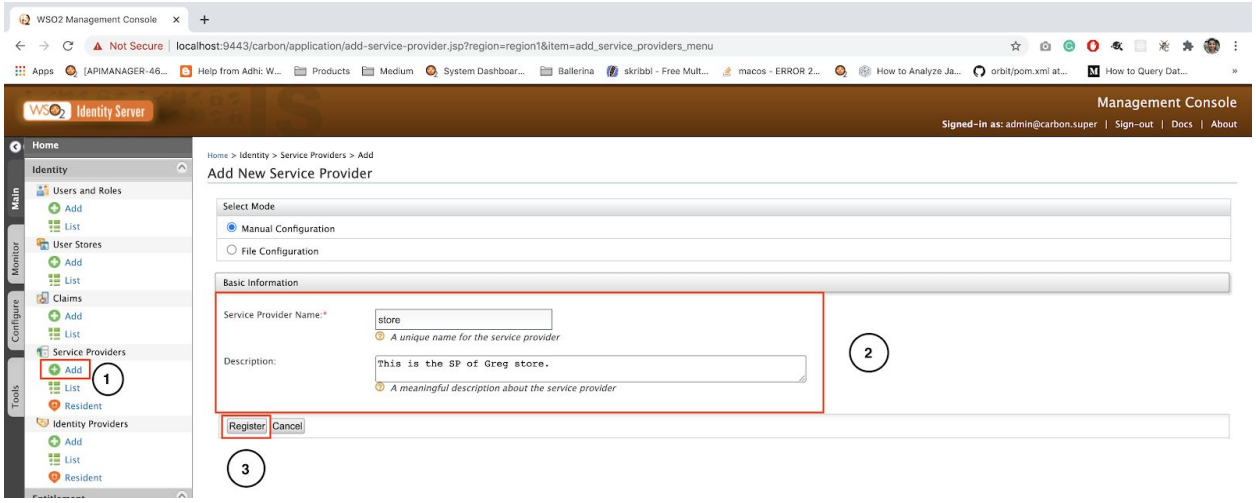

Fig 11: Create publisher SP and register

c. Service Provider details will be displayed and check the "**SaaS Application**" if multi-tenancy is used.

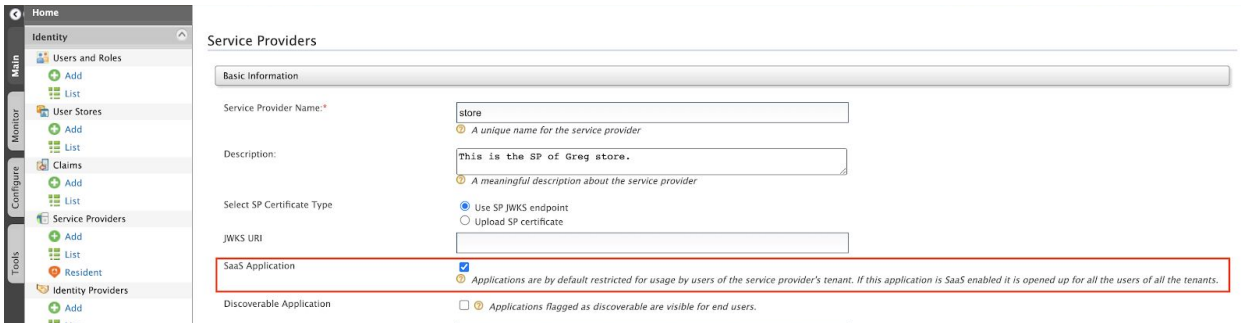

Fig 12: Enable “SaaS Application”

d. Click on **Inbound Authentication Configuration**, next click **SAML2 Web SSO Configuration** and then click **Configure**.

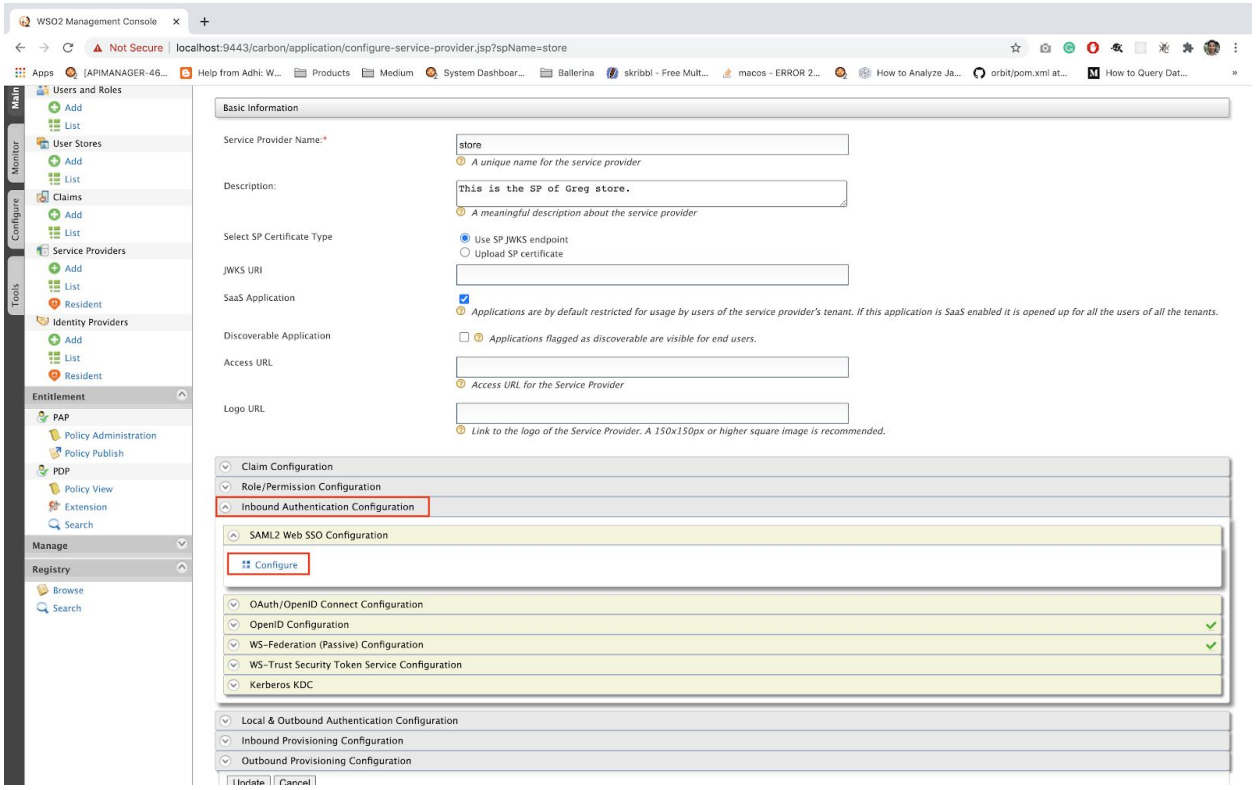

Fig 13: Configure SAML SSO

e. Provide the following details to the created store SP.
   i. **Issuer:** Enter store for this. Note that the service provider name you provide here should be the same as the issuer value configured in **<wso2greg-5.4.0>/repository/deployment/server/jaggeryapps/store/config/store.json** file.

ii. **Assertion Consumer URLs:** Enter the URL of the Store node, and click Add.

- Since the public certificates of Greg and IS were not added to each other we have unchecked the "**Enable Response Signing**", "**Enable Signature Validation in Authentication Requests and Logout Requests**", and "**Enable Assertion Encryption**".

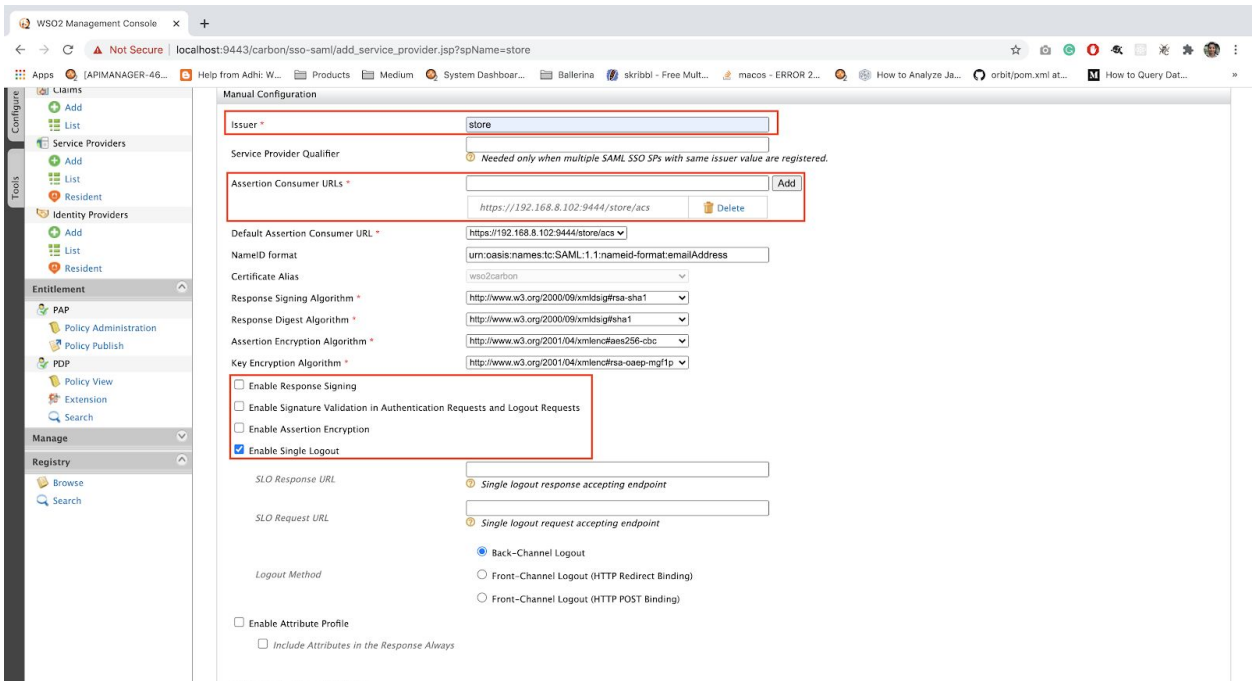

Fig 14: Configure store SP

f. Click Register.

Fig 15: Register the store SP

g. Click on **Local & Outbound Authentication Configuration** and check **Use tenant domain in local subject identifier.**

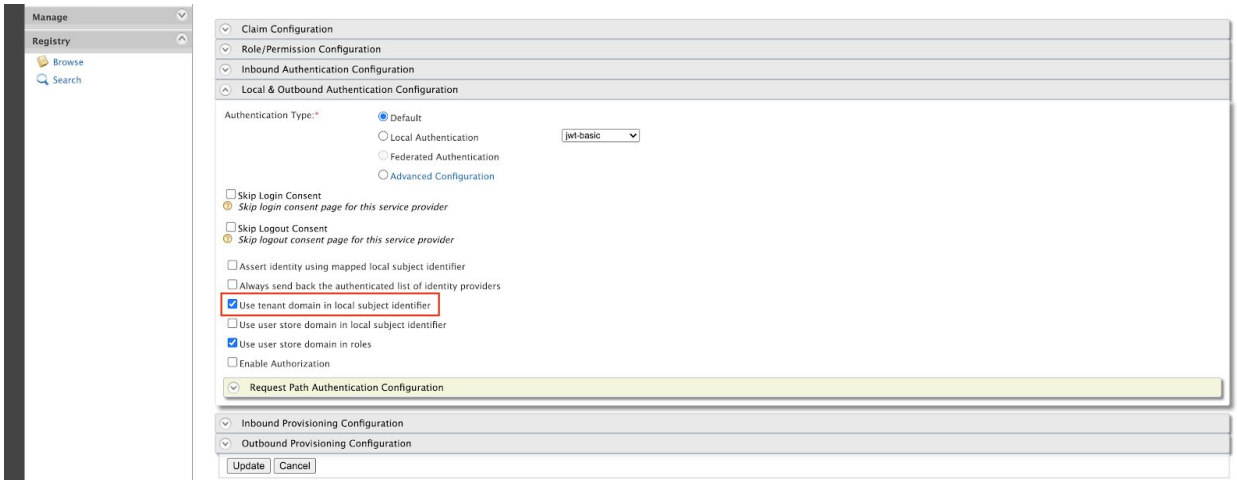

Fig 16: Configure Outbound and Authentication

h. Update the store SP configurations.

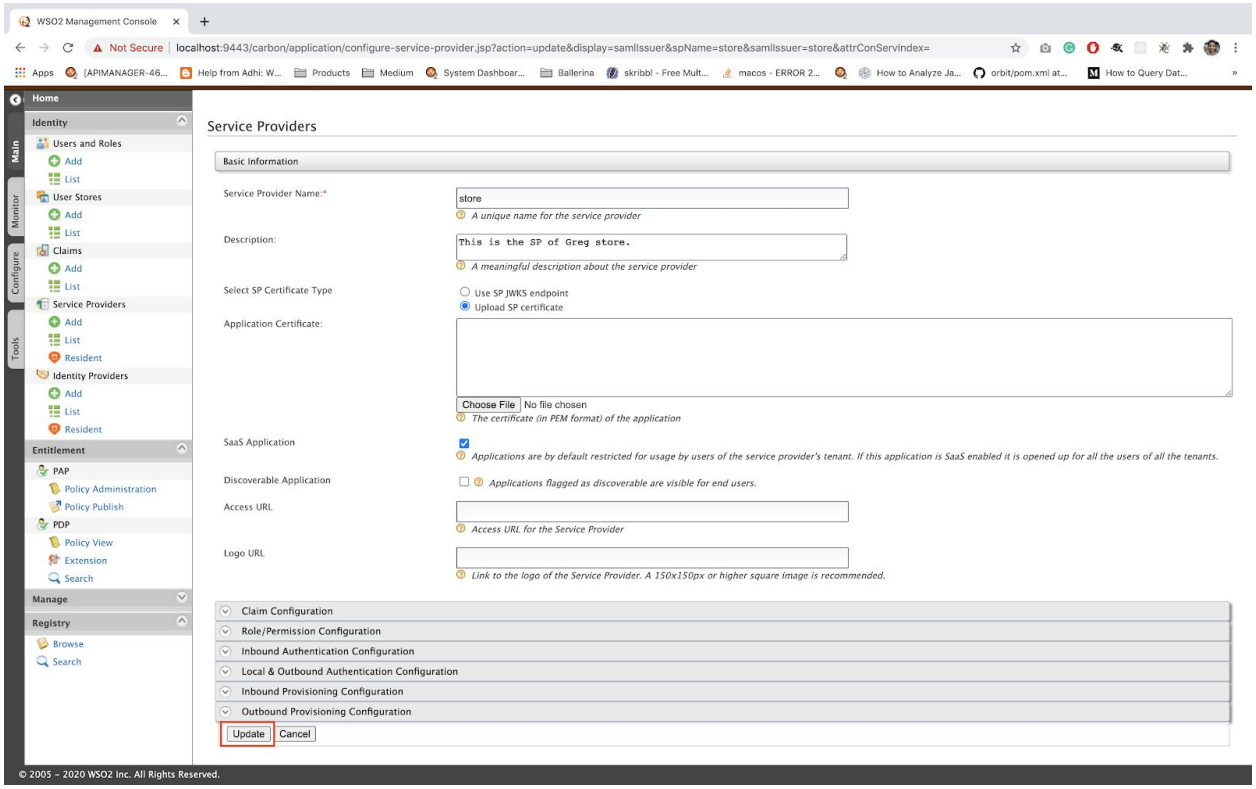

Fig 17: Update the store SP configurations

i. The created store SP will be listed as below.

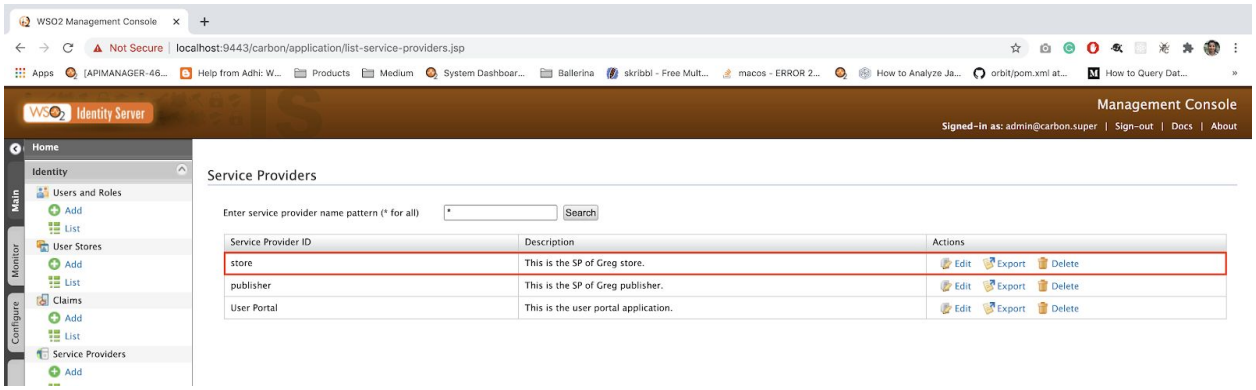

Fig 18: List the store SP

Tip:

In order to confirm whether the above configurations are correct, you can login to Greg carbon management console as a super tenant admin and create a user. If you can see the same user in the IS carbon management console the flow is correct. This is applicable wise versa.

Setting up multi-tenants environment in WSO2 GReg and WSO2 IS

Next, we need to set up the multi-tenants environment in both WSO2 GReg and WSO2 IS. The following steps can be followed to achieve this.

WSO2 GReg multi-tenants environment setup

1. Login to GReg carbon management console as super tenant admin user (Default credentials - username: admin, password: admin)

2. Navigate to **Configure → Multitenancy → Add New Tenant → Add the tenant details → Save**

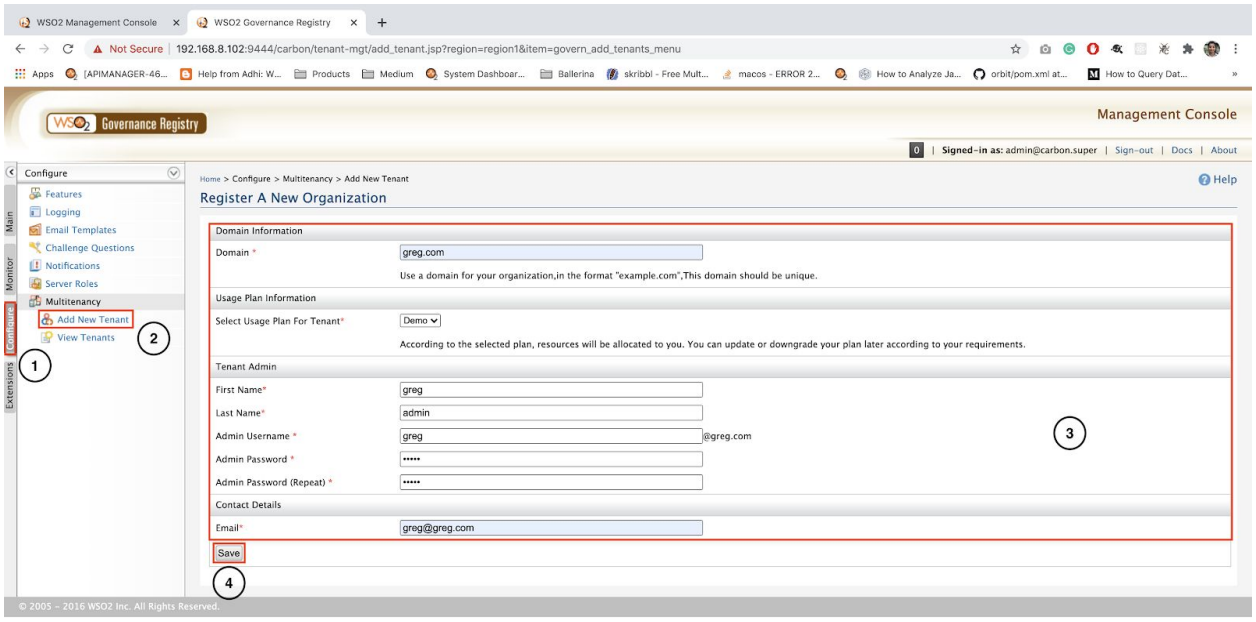

Fig 19: Add new tenant in GReg

3. The tenant will be created successfully.

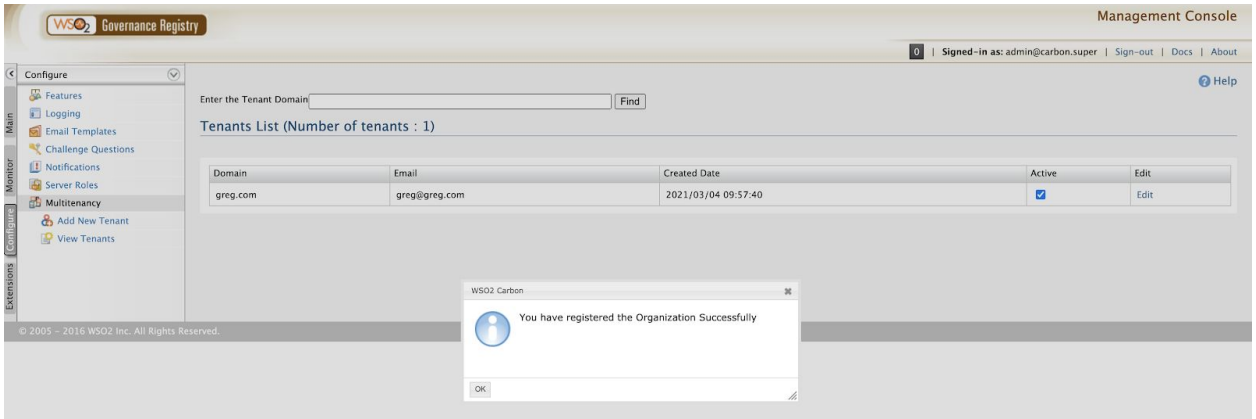

Fig 20: New tenant was created successfully in GReg

4. Login as the newly created tenant admin user to GReg carbon management console (E.g: username: greg@greg.com, password:admin)

5. Create a new tenant user in greg.com tenant as follows by navigating to **Main** → **Identity** → **Users and Roles** → **Users** → **Add**. Please note that the user is created in the PRIMARY domain.

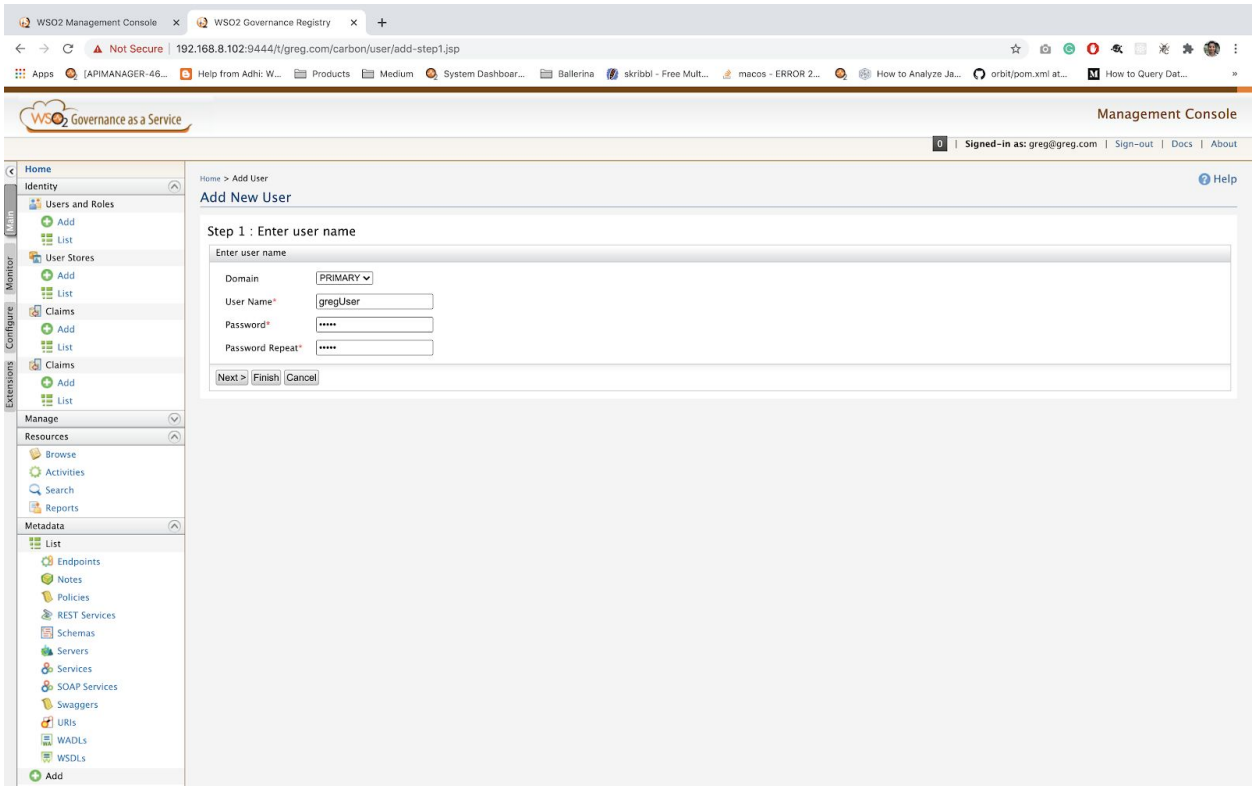

Fig 21: Create new tenant user in greg.com tenant in GReg

6. Click on Finish and the user will be created successfully.

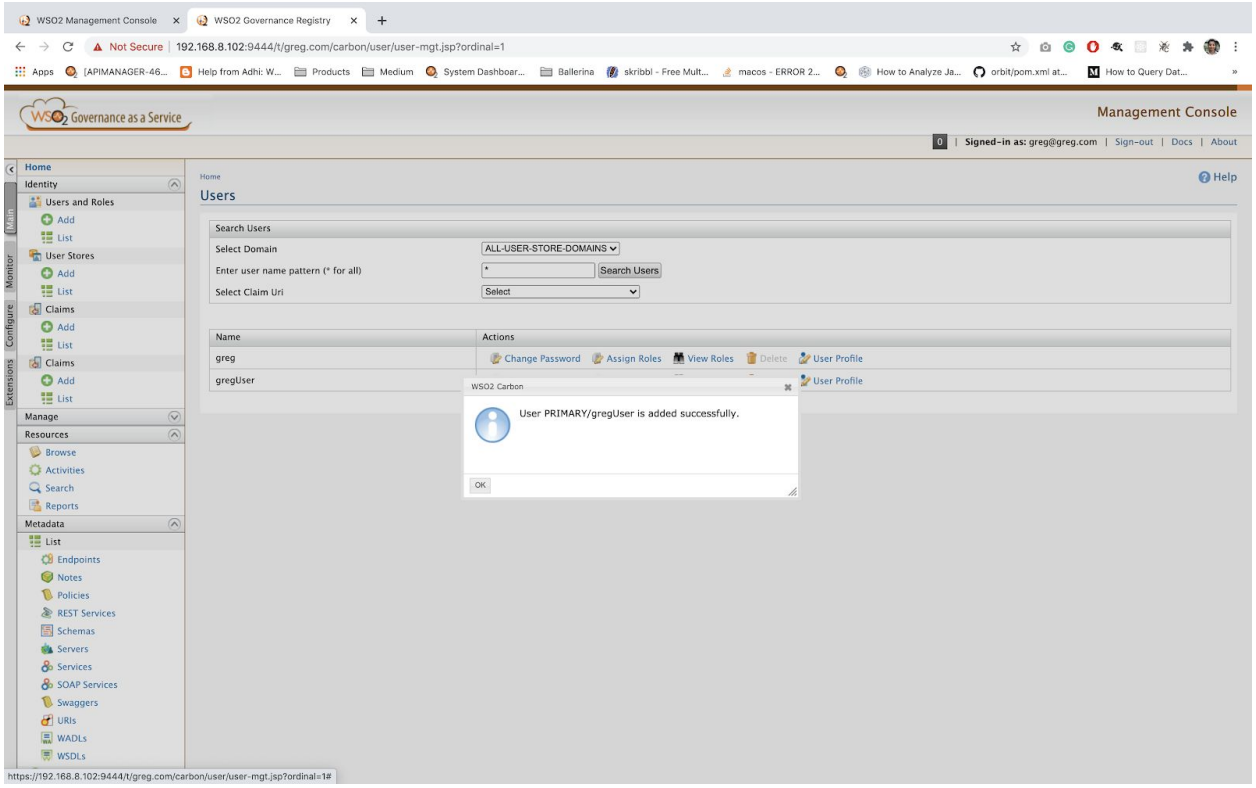

Fig 22: Successfully created the new tenant user in greg.com tenant in GReg

7. Create a publisher role in the PRIMARY domain as the primary user store is being shared between GReg and IS. Navigate to **Main → Identity → Users and Roles → Roles → Add.**

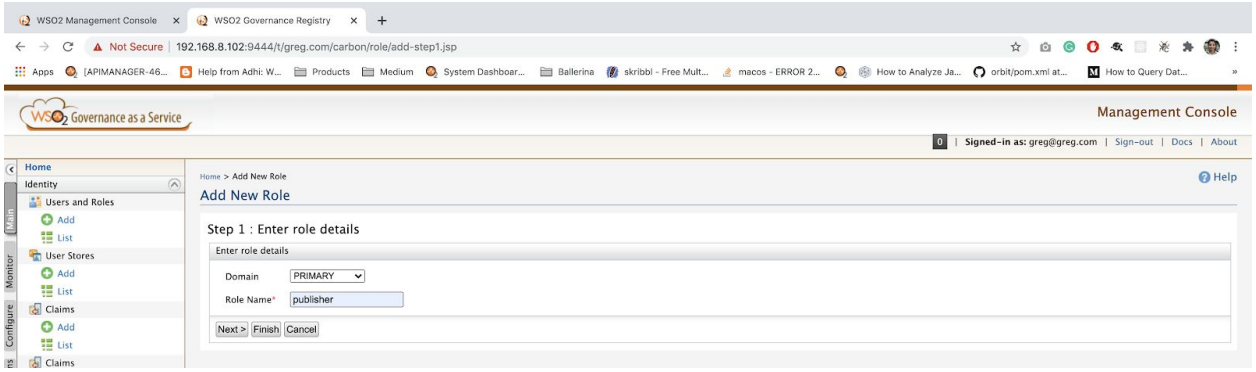

Fig 23: Create the publisher role in greg.com tenant in GReg

8. When you click **Next,** you will be able to assign the permissions for the publisher role as follows.

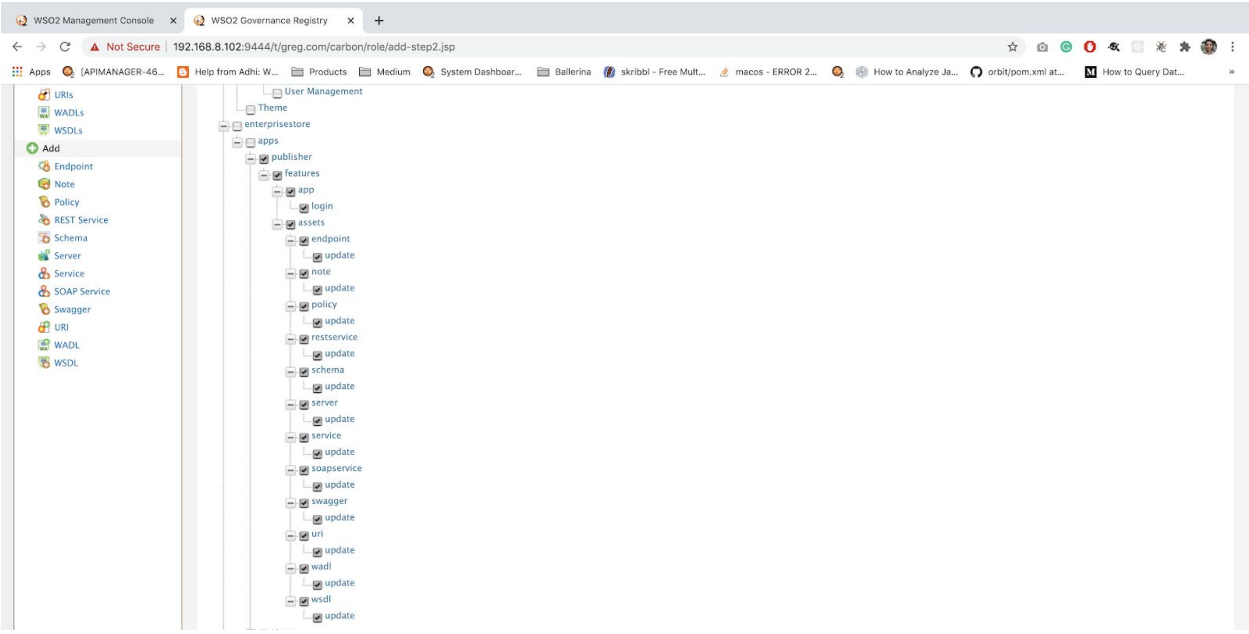

Fig 24: Add publisher role permissions - 1

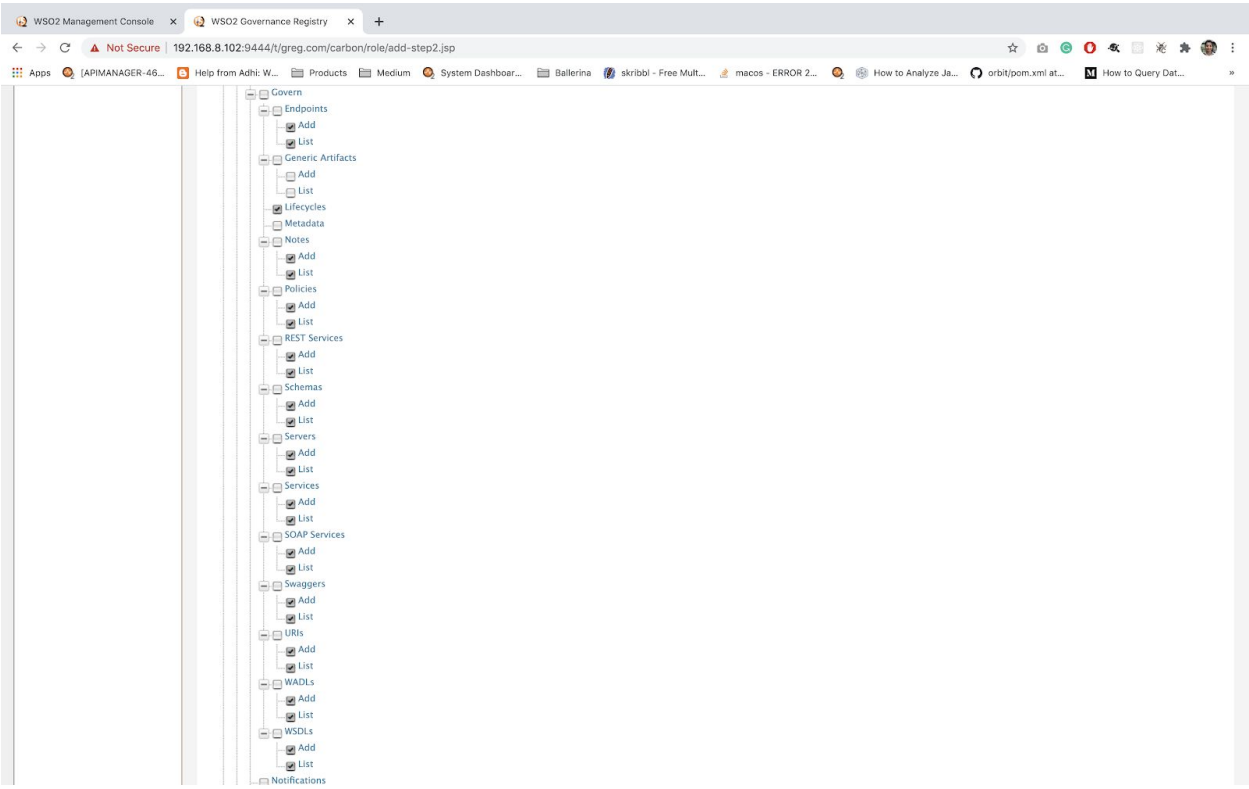

Fig 25: Add publisher role permissions - 2

9. Click on the **Next** button.

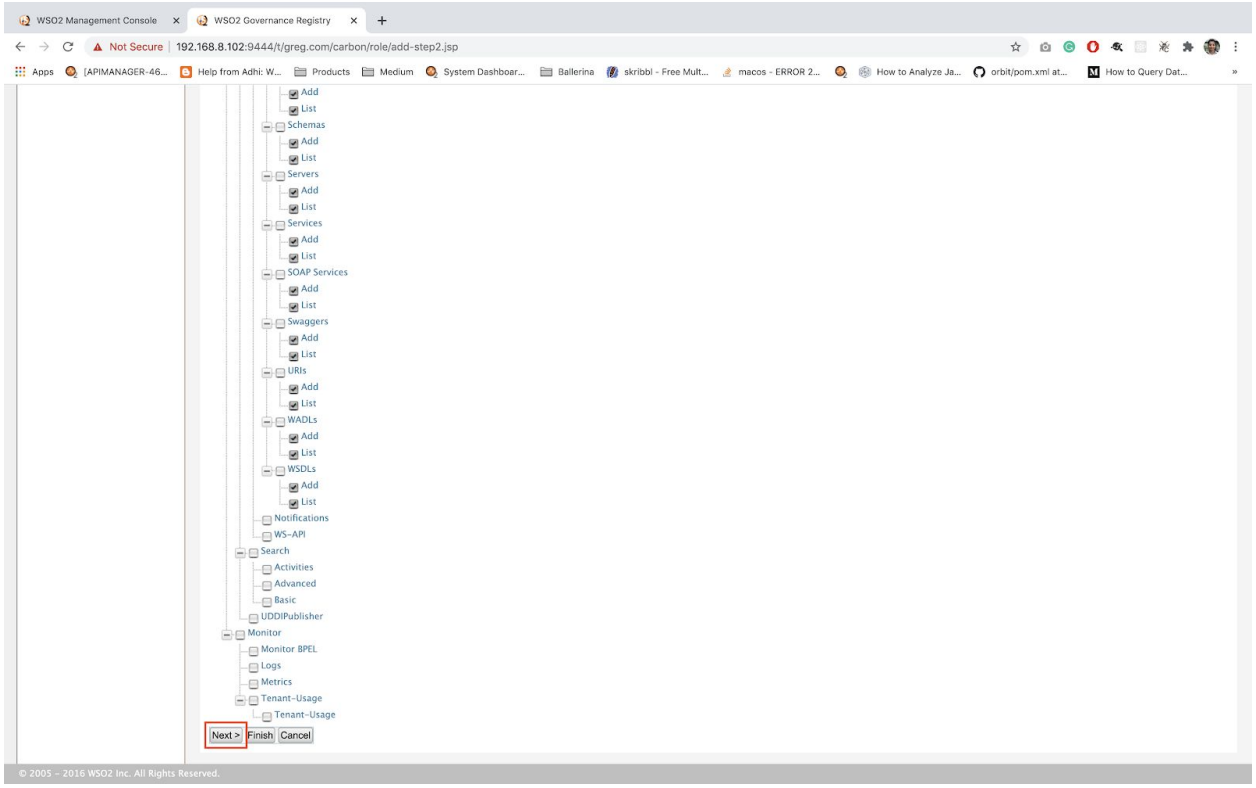

Fig 26: Click on the **Next** button

10. Assign the publisher role to gregUser which was created under the greg.com tenant and click on **Finish**.

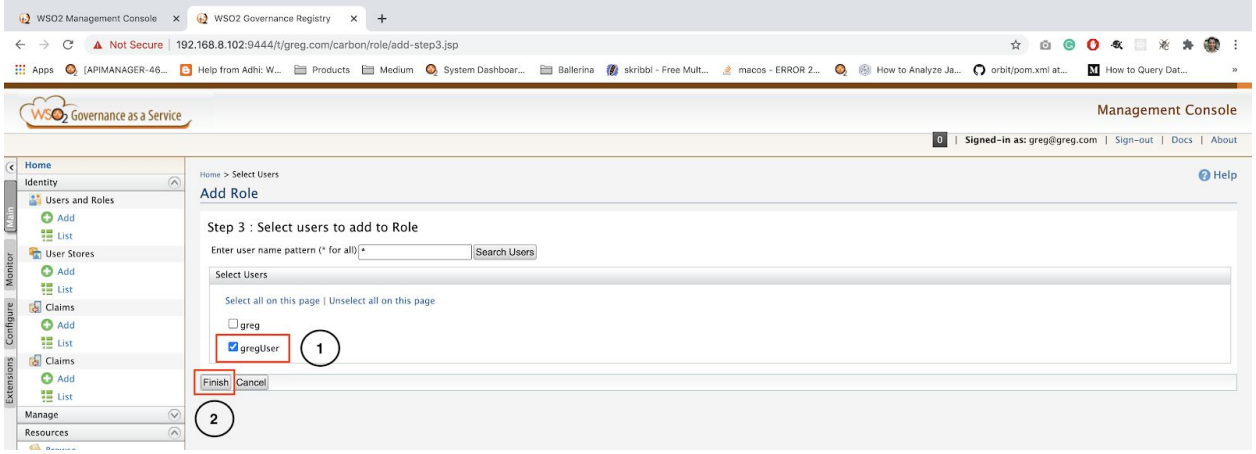

Fig 27: Assign user to publisher role

Tip:

Since the GReg has various assets the permissions available in GReg and IS carbon management consoles are different. Therefore, the roles should be created in the GReg side and assign the roles to the relevant users.

WSO2 GReg multi-tenants environment setup

By default, the tenants created in the GReg will not be visible in the IS carbon management console. Therefore. You need to create the same tenant as in the GReg carbon management console in the IS carbon management console.

1. To check this, log in to the IS carbon management console as the super tenant admin user (Default credentials - username: admin, password:admin).

2. Next, navigate to **Configure → Multitenancy → Tenants will not be available.**

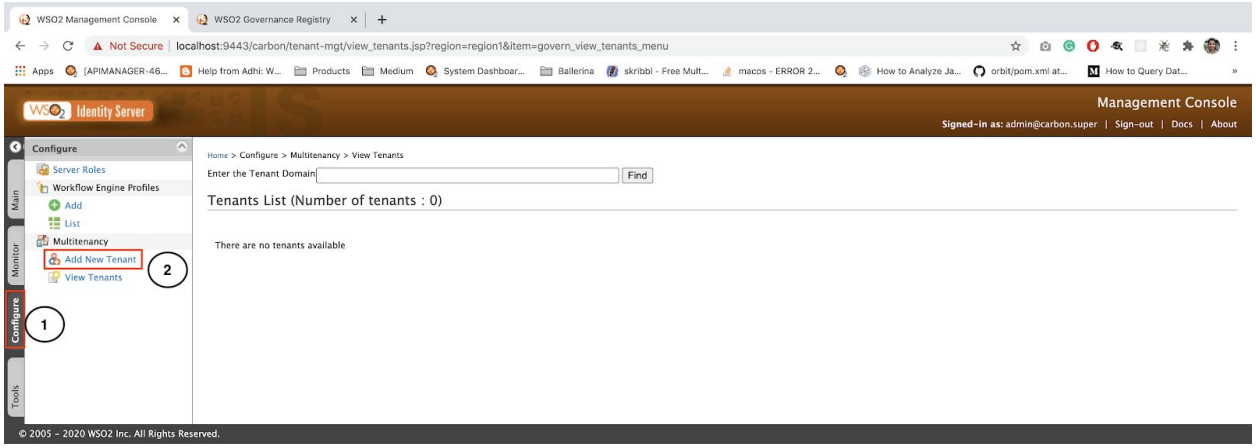

Fig 28: No tenants are available in the IS carbon management console

3. Navigate to **Configure → Multitenancy → Add New Tenant → Add the tenant details → Save**

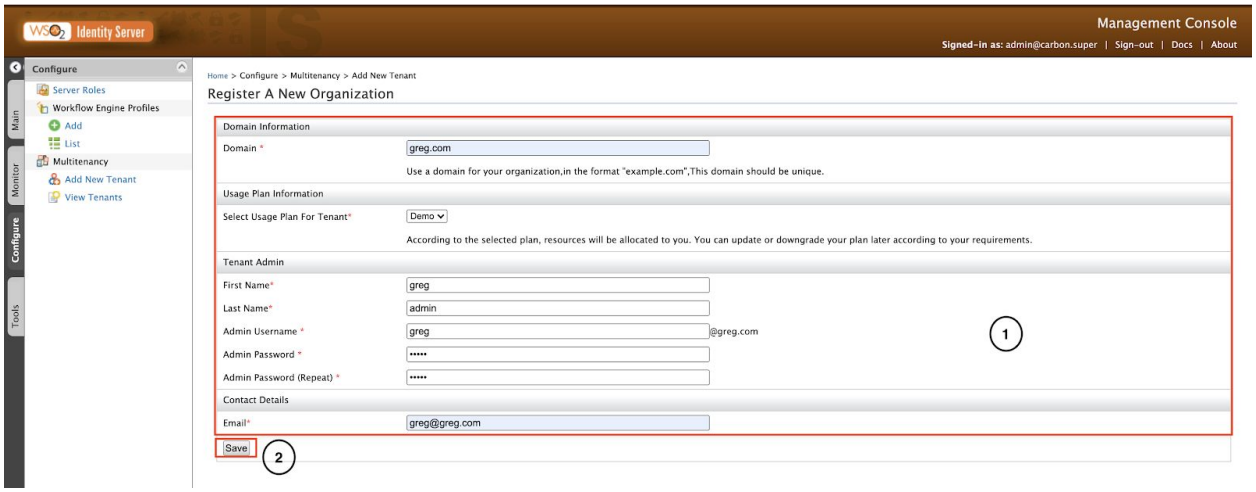

Fig 29: Create the same tenant as created in GReg and save

4. The tenant will be created successfully in the IS carbon management console.

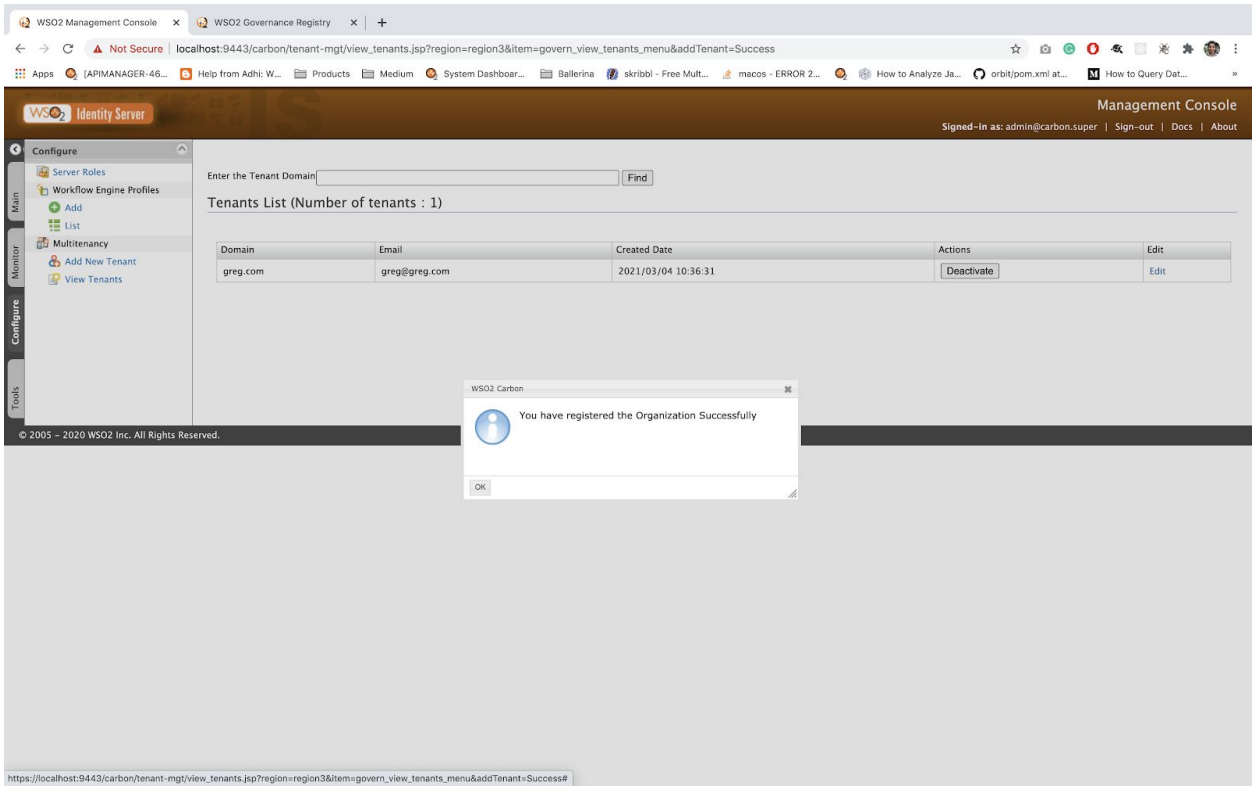

Fig 30: Successfully created the tenant in IS

Now when you check the users in the greg.com tenant created in IS you will be able to see the users created in the same tenant in GReg.

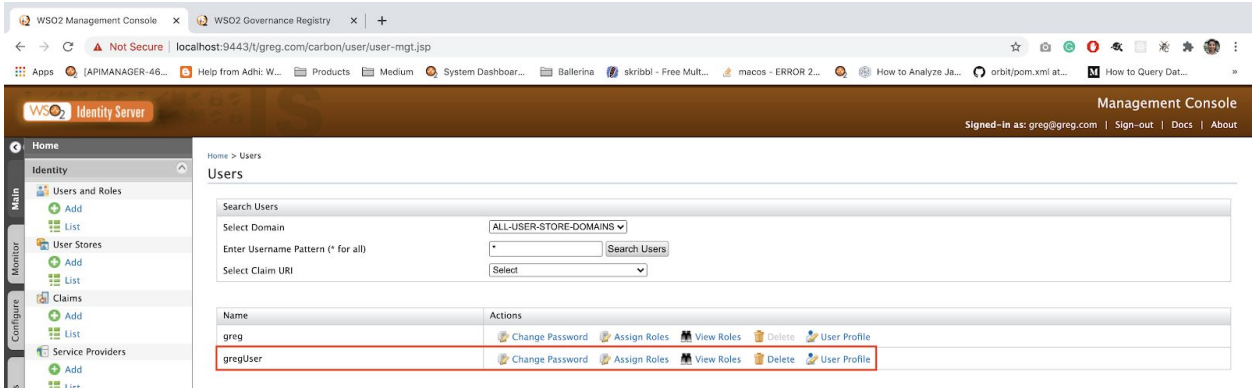

Fig 31: Tenant users in IS

Try out the SAML SSO flow

Let's try to login to the GReg publisher via **https://<GReg host or IP address>:9444/publisher** and it will navigate you to the IS login page as follows.

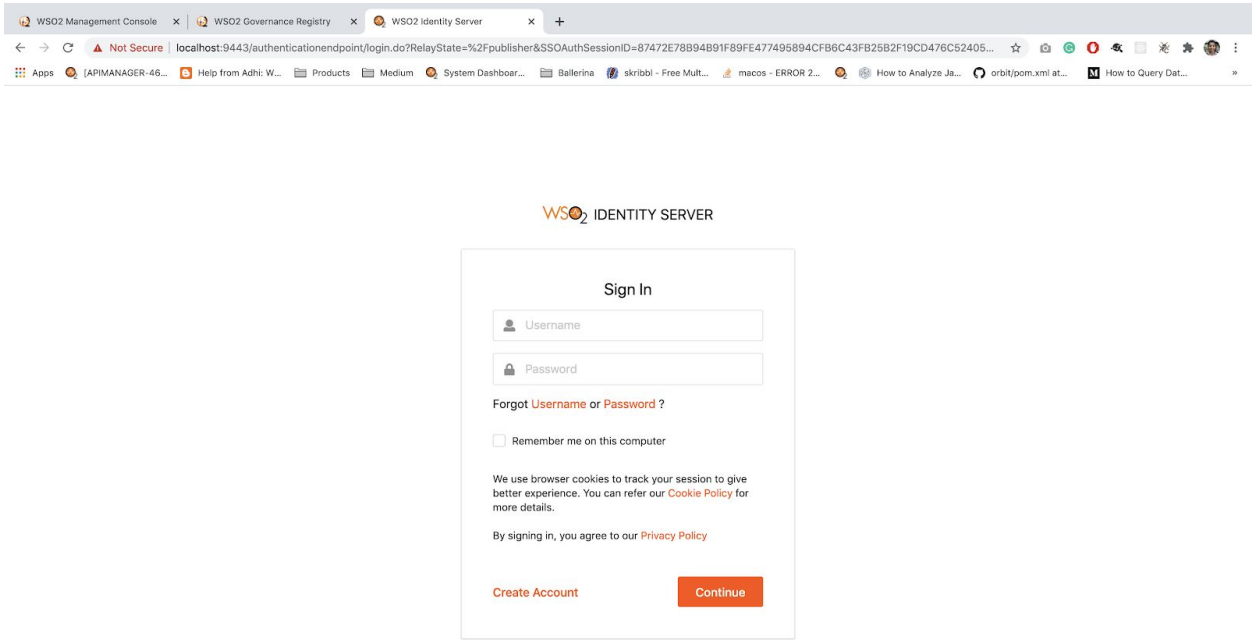

Fig 32: IS login page when trying to access the GReg publisher login page

Provide the greg.com tenant user, gregUser's credentials.

- Username: gregUser@greg.com
- Password: admin

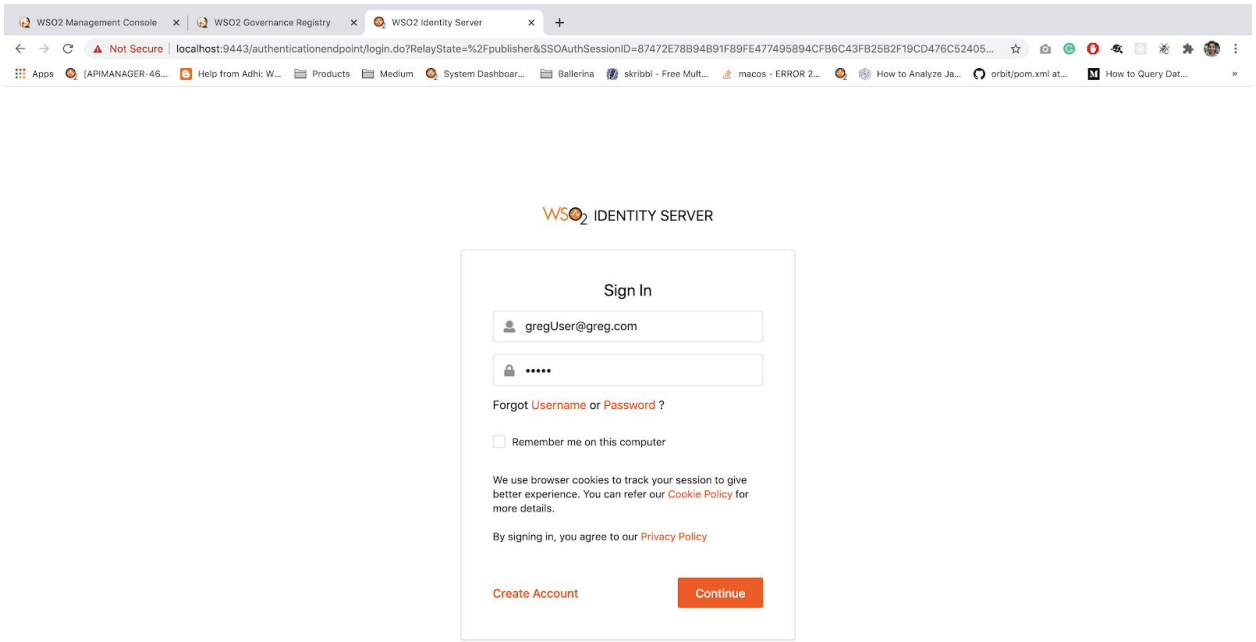

Fig 33: Provide the login credentials

You will be able to login as gregUser@greg.com in the GReg publisher as follows.

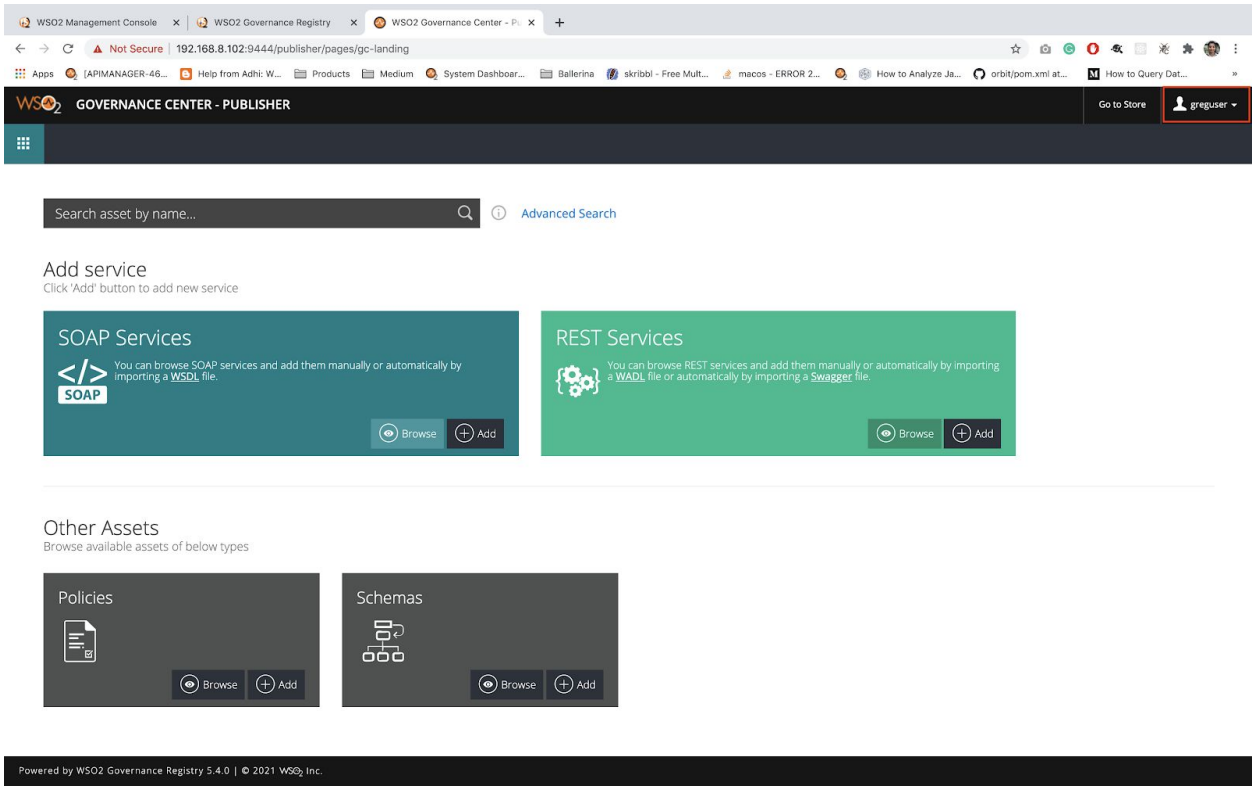

Fig 34: Successfully logged into GReg publisher portal

Similarly, you can login to the GReg store portal as well.

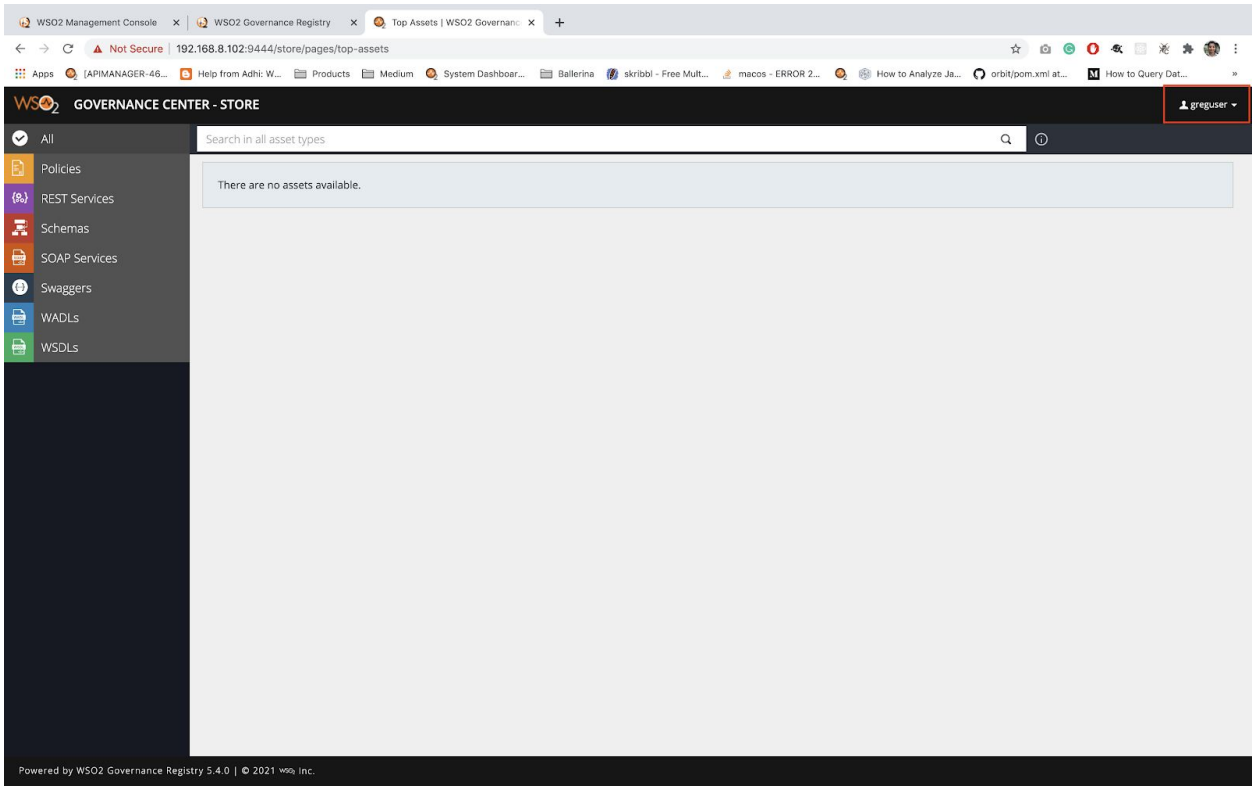

Fig 35: Successfully logged into GReg store portal

References

[1] Configure LDAP as the primary user store in GReg

[2] https://docs.wso2.com/display/Governance540/Configuring+Single+Sign-on