Design Room ONE

Setting up Authentication in Design Room ONE with Keycloak

This document describes how to setup and use authentication and user management in Design Room ONE by means of integrating with Keycloak.

Table of Contents

Keycloak Server Installation

The instructions below use the following variables that need to be replaced with their actual values
DRONE_HOST_NAME e.g. drone.mycomp.any
KEYCLOAK_HOST_NAME e.g. keycloakhost.mycomp.any
KEYCLOAK_IP_ADDRESS e.g. 10.20.30.40
KEYCLOAK_INSTALL_DIR e.g. C:\Install\Keycloak
If Keycloak and Design Room ONE are installed on different machines, these machines should be able to communicate with HTTP/HTTPS requests. This usually means adjusting firewall settings to allow such communications and ensuring all host names can be successfully resolved on every machine.

Downloading Keycloak

Keycloak can be downloaded from the following site:
https://www.keycloak.org/downloads
Check system_requirements.pdf document for supported versions.
Unzip the downloaded file into an installation directory of your choice. We will refer to this installation directory as **KEYCLOAK_INSTALL_DIR**.

Keycloak Server Configuration – Quarkus

This section is only applicable to Quarkus distribution of Keycloak.

Configuring port and hostname

By default, Keycloak will use **8080** as http port and **8443** as https port and **localhost** as host. To use custom values, the following commands can be used.
To configure http port:

`http-port=<port>` --- **KEYCLOAK_INSTALL_DIR/conf/keycloak.conf**
`--http-port <port>` --- command line

To configure https port:

`https-port=<port>` --- **KEYCLOAK_INSTALL_DIR/conf/keycloak.conf**
`--https-port <port>` --- command line

To configure hostname:

`hostname=<keycloak.mycomp.any>` ---**KEYCLOAK_INSTALL_DIR/conf/keycloak.conf**
`--hostname <keycloak.mycomp.any>` --- command line

e.g.,

```
kc.bat start --hostname <keycloak.mycomp.any> --http-port 8051
```

Refer this link for more.

Initial Startup of Keycloak Server

Run the standalone version of the server in **KEYCLOAK_INSTALL_DIR/bin**

For Linux

```
kc.sh start --auto-build
```

For Windows

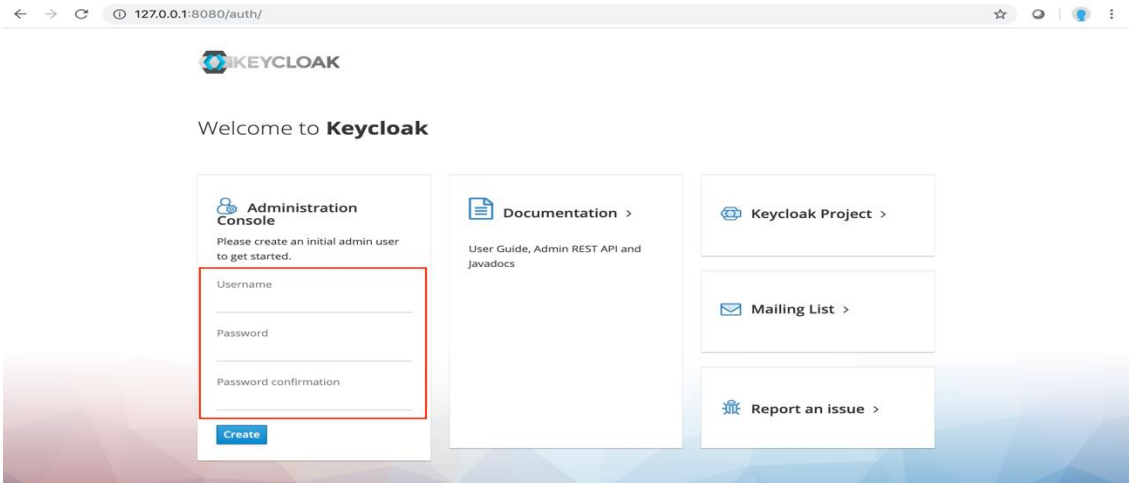
```
kc.bat start --auto-build
```

Creating a Realm Administrator

Note: This step must be performed on a machine running Keycloak.

Open a browser and navigate to https://KEYCLOAK_HOST_NAME:8443/ (or HTTP endpoint).

Note: the browser may show a warning if e.g., certificate does not match the host name



Specify the admin credentials and press Create.

This admin user is a super user with all full access to Keycloak (realm creation, update, deletion, user creation, update, deletion, etc.)

Starting Keycloak Server

Stop the Keycloak server and start it again with `--hostname` parameter.

Run the standalone version of the server in **KEYCLOAK_INSTALL_DIR/bin**

For Linux,

```
kc.sh start --auto-build --hostname KEYCLOAK_IP_ADDRESS
```

For Windows,

```
kc.bat start --auto-build --hostname KEYCLOAK_IP_ADDRESS
```

It is possible to use 0.0.0.0 instead of **KEYCLOAK_IP_ADDRESS** to allow connection from any interface, by default only connections from the same machine will be accepted. See Keycloak documentation corresponding to the version specified in system_requirements.pdf for details.

Importing Previous Configuration

If you configured drone realm in version earlier than 2.3 of Design Room ONE you need to perform the following steps to keep your configuration, i.e., users and roles. If you are doing a fresh installation of Design Room ONE proceed with the next section.

1. Export your current realm data
    a. Ensure your keycloak server instance is stopped
    b. Run the below command to export your current drone Keycloak realm data

    Run the command below to export realm data to a directory:

    For Linux

    ```
    KEYCLOAK_INSTALL_DIR/bin/kc.sh export --dir <directory>
    ```

    For Windows

    ```
    KEYCLOAK_INSTALL_DIR\bin\kc.bat export --dir <directory>
    ```

    To export to a single file, use below command:

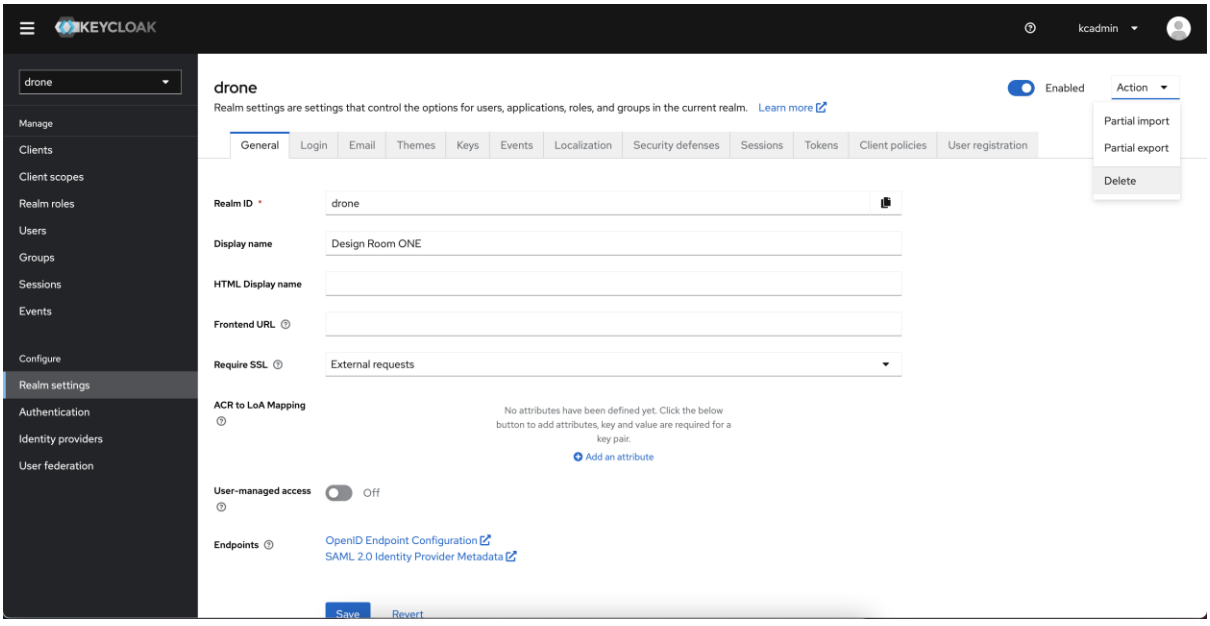
    For Linux

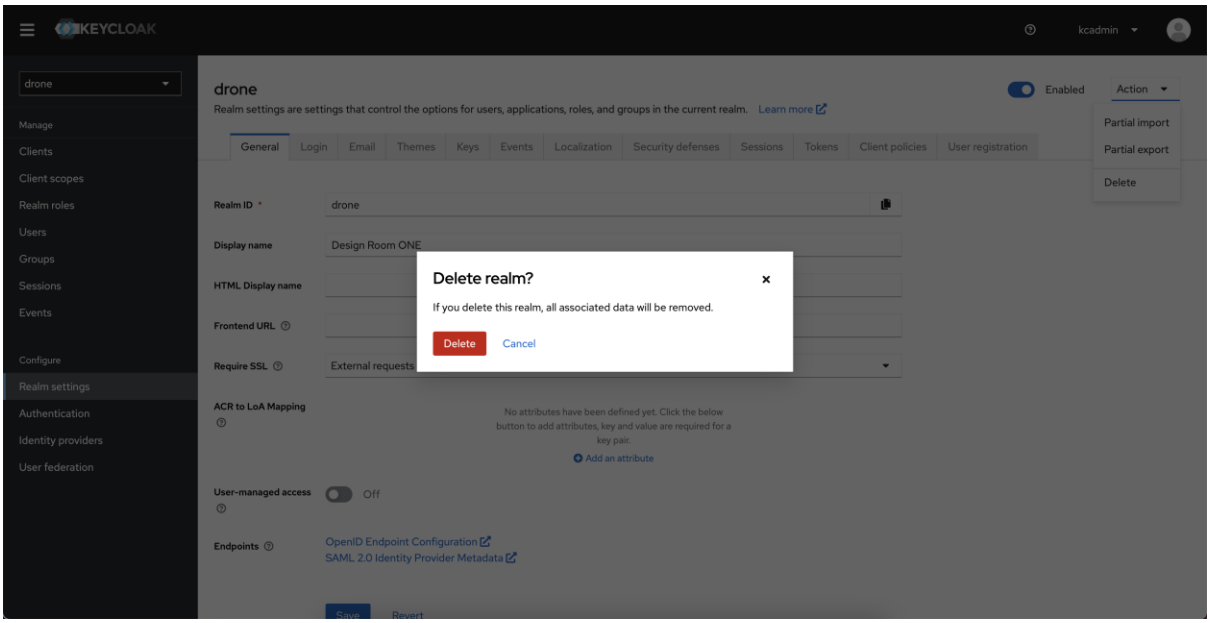
    ```
    KEYCLOAK_INSTALL_DIR/bin/kc.sh export --file <file>
    ```

    For Windows

    ```
    KEYCLOAK_INSTALL_DIR\bin\kc.bat export --file <file>
    ```

2. Delete existing drone realm
    a. Start your Keycloak server instance
    b. Choose the **Realm** (i.e. drone) in the dropdown on the left panel.
    c. Navigate to “Realm Setting” and then click on **Actions** -> **Delete** to DRONE realm as shown below.





3. Import Design Room ONE provided sample realm
   Run the following command
   For Linux
   `KEYCLOAK_INSTALL_DIR/bin/kc.sh import --file DR_ONE_INSTALL_DIR/DR_Install/Resources/Keycloak/drone-realm-export.json`
   For Windows
   `KEYCLOAK_INSTALL_DIR\bin\kc.bat import --file DR_ONE_INSTALL_DIR/DR_Install/Resources/Keycloak/drone-realm-export.json`

```
2022-07-06 18:14:15,665 INFO  [org.keycloak.exportimport.util.ImportUtils] (main) Realm 'drone' imported
2022-07-06 18:14:15,701 INFO  [org.keycloak.services] (main) KC-SERVICES0032: Import finished successfully
2022-07-06 18:14:16,145 INFO  [io.quarkus] (main) Keycloak 18.0.0 on JVM (powered by Quarkus 2.7.5.Final) started in 9.666s.
```

4. Import your exported realm
   Run the command below to import from a directory:
   For Linux
   `KEYCLOAK_INSTALL_DIR/bin/kc.sh import –dir <directory>`
   For Windows
   `KEYCLOAK_INSTALL_DIR\bin\kc.bat import –dir <directory>`

   Run the command below to import from a single file:
   For Linux
   `KEYCLOAK_INSTALL_DIR/bin/kc.sh import –file <file>`
   For Windows
   `KEYCLOAK_INSTALL_DIR\bin\kc.bat import –file <file>`

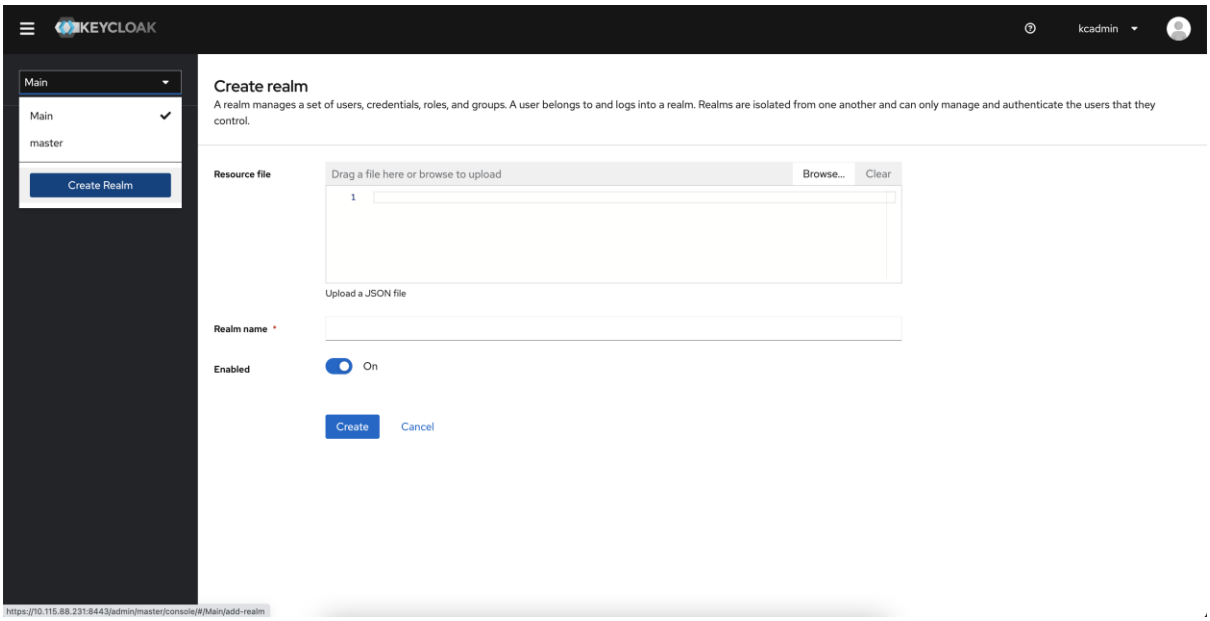
5. Start the Keycloak server, see this section for details.
6. Ensure that all previously created users have **view-realm** role as explained in creating users section
7. Realm configuration is now complete, you should skip instructions in the next section and proceed with Setting up Users and Roles.

Refer Importing and Exporting Realms for more details.

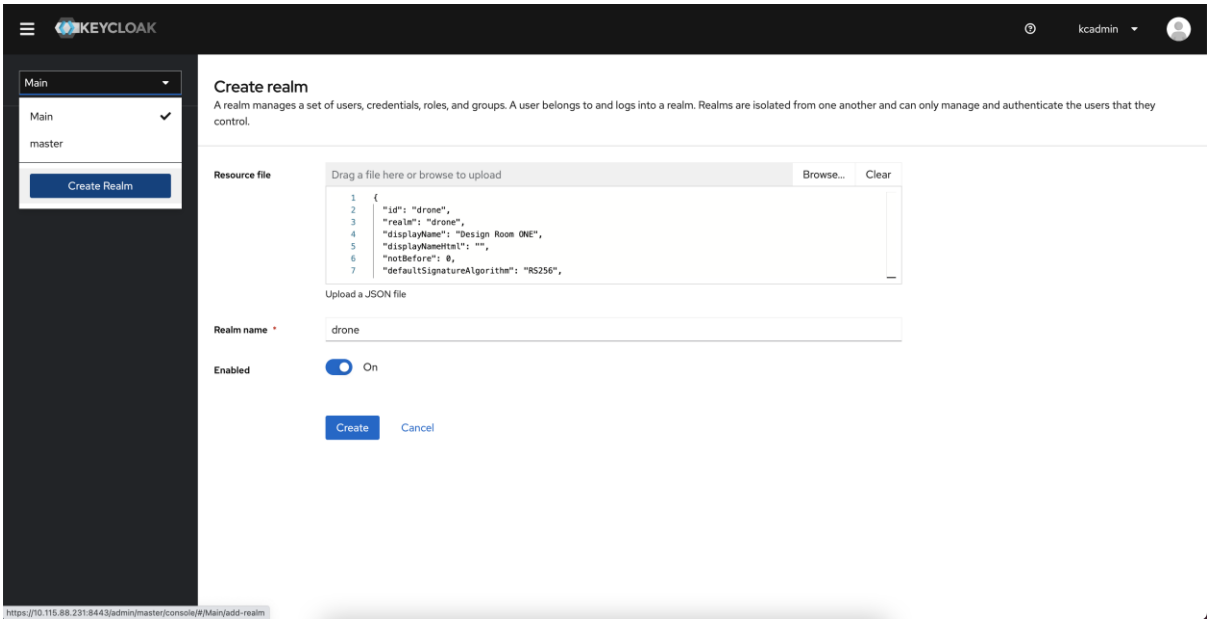
Configuring the Realm

Note: this step is easier to do from the machine where Design Room ONE is installed

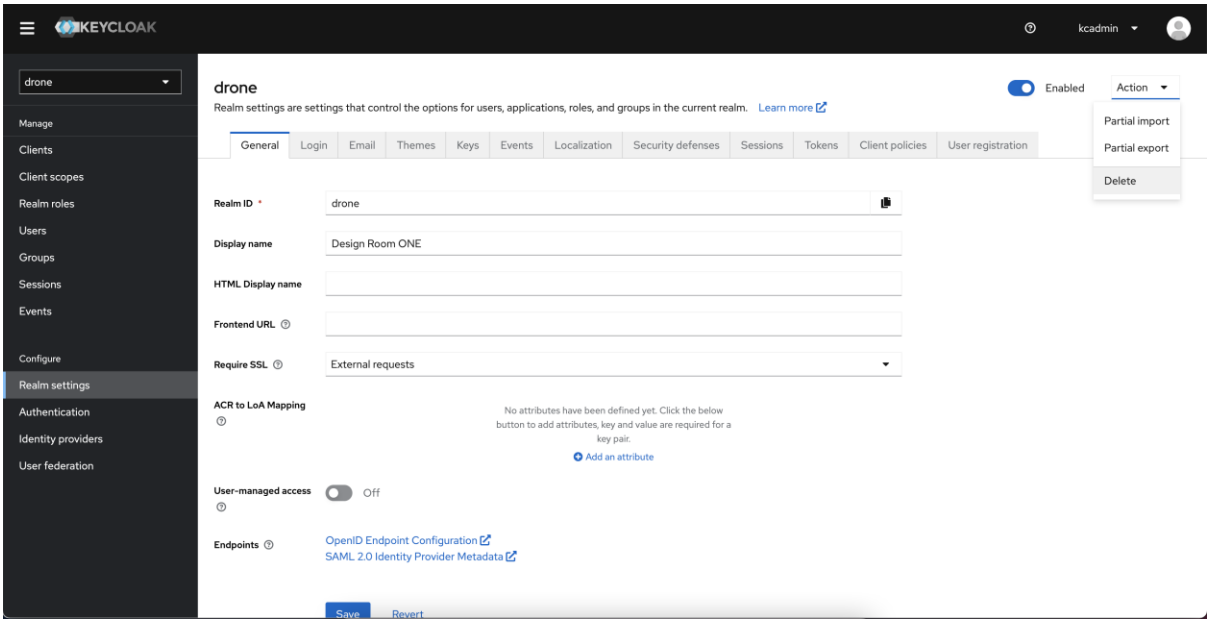
1. Import the realm data
    a. Hover over **Realm** dropdown in top left corner and click on **Create Realm** and wait for it load to the below page.



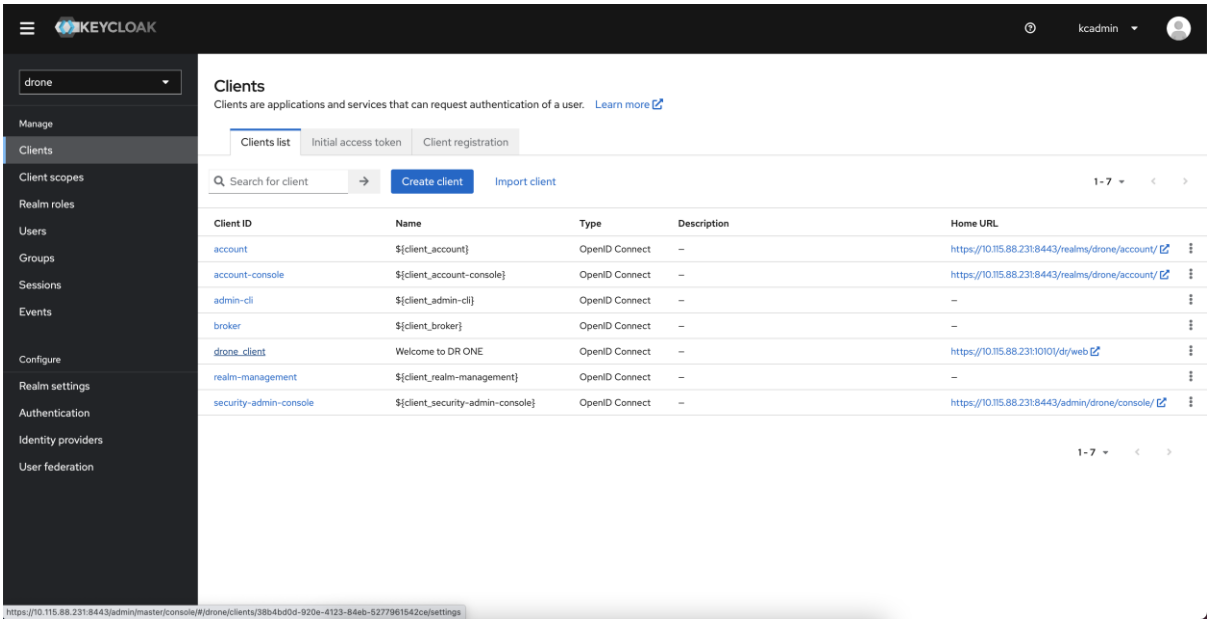
    b. Click on **Select File** and point to **drone-realm-export.json** file. It can be found in machine where **Design Room ONE** is installed under following path: DR_ONE_INSTALL_DIR/DR_Install/Resources/Keycloak/drone-realm-export.json
    c. Then, click **Create**



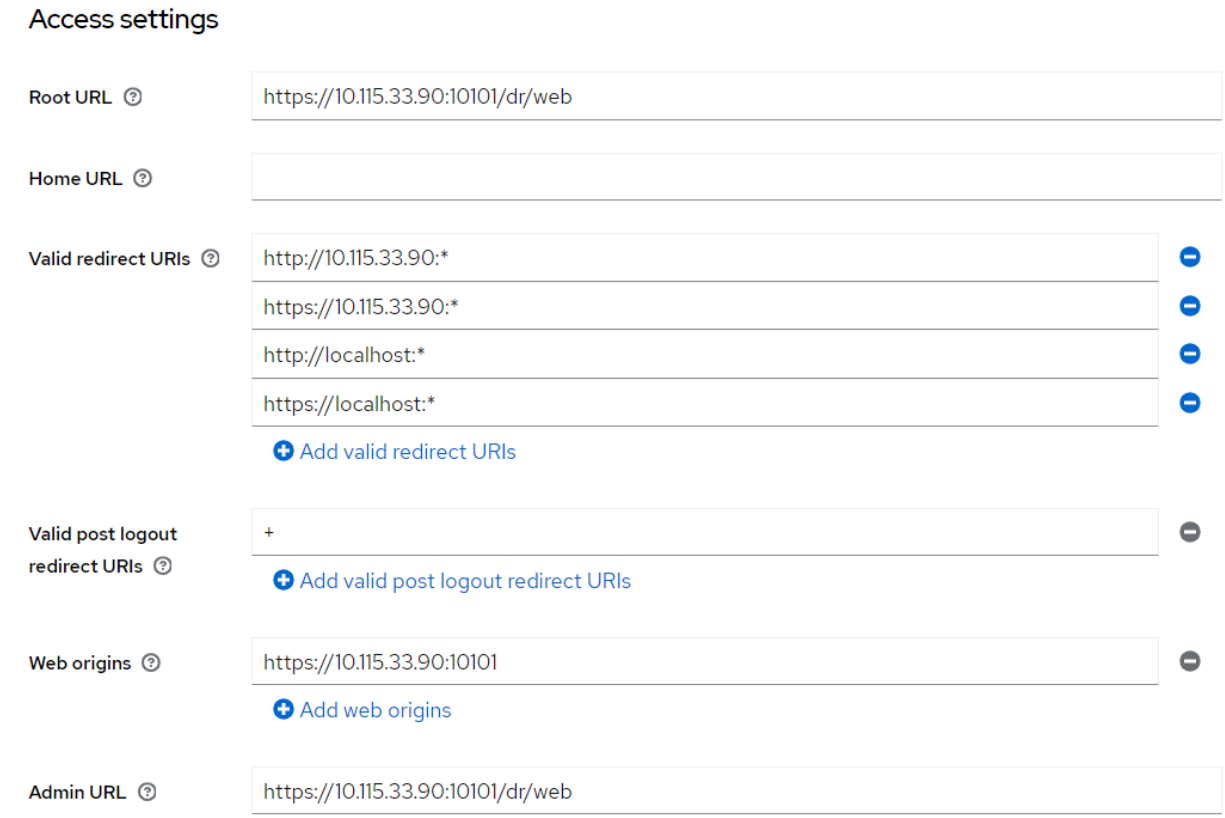
d. Your **Drone** realm should be created successfully as shown below:



2. Setup drone_client
    a. Under **Clients** click on the drone_client link

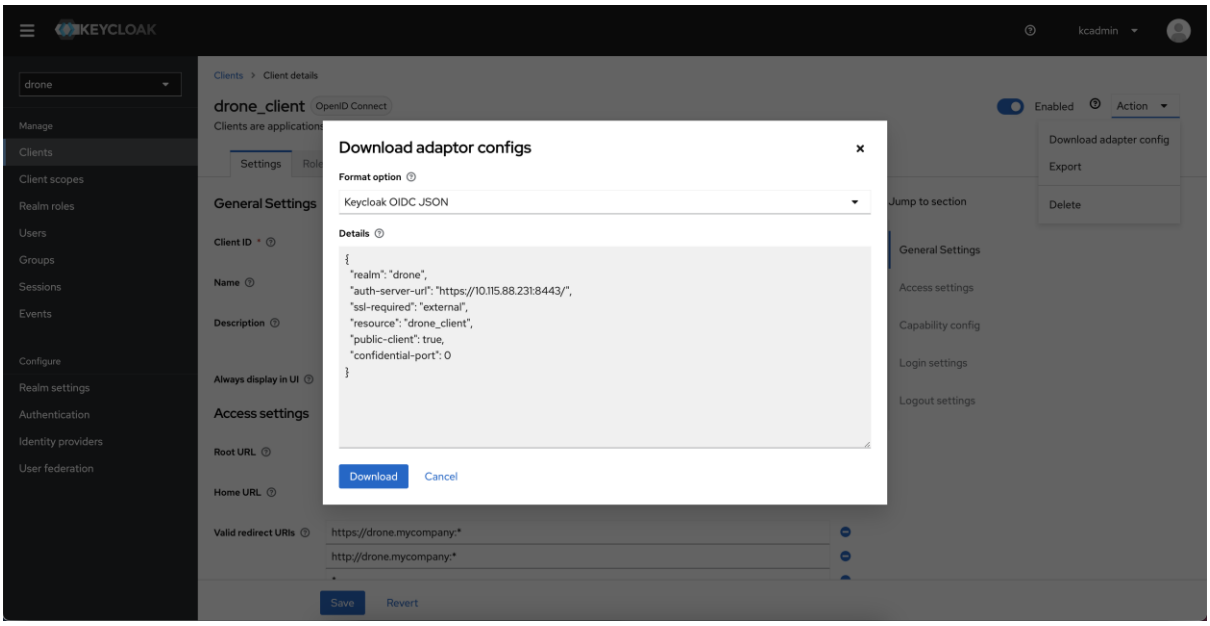


b. Click on the **Settings** top menu, scroll down and add **DRONE_HOST_NAME** (or ip address) accessible by other machines for the server Keycloak as valid redirect URIs as in example below. It is recommended to use lowercase letters.



c. Make sure **Root URL**, **Admin URL**, and **Web Origins** fields are also updated with **DRONE_HOST_NAME** and hit **Save**

d. On the top right **Action** dropdown, select **Download adaptor config** menu.
e. In the **Download adaptor configs** popup, select **Keycloak OIDC JSON** in **format option** and click **Download**.



Note: Copy and paste **auth-server-url** and **ssl-required** properties in the file **DR_ONE_INSTALL_DIR/OnPrem_Design_Room/config/server-config.json** under **dr_keycloak_config** object. Also change **"dr_auth"** attribute value to **"keycloak"**. Then the configuration file would look like this.

```
  // Authentication (for accessing information stored in Design Room ONE)
  // "none": Do not use any authentication. Everyone can access all designs.
  // "jazz": Use Jazz authentication. User needs to be logged in to Jazz to acces
s designs.
  // "keycloak": Use Keycloak authentication. User needs to be logged via keycloa
k to access designs.
  "dr_auth": "keycloak",

  //Keycloak configuration
  "dr_keycloak_config": {
    "auth-server-url": "https://drone.mycomp.any:8443/",
    //Defines security level for Keycloak server
    //"none": HTTPS not required for any IP address
    //"external": Private IP and localhost can access without HTTPS
    //"all": HTTPS required for all IP addresses
    "ssl-required": "external"
  }
```

Note: Make sure correct **KEYCLOAK_HOST_NAME** (or ip-address) specified in **auth-server-url** attribute, and the URL is accessible by machine where Design Room ONE server is installed and from user machines. It is recommended to use lowercase letters in the URLs.

Keycloak Server Configuration – Wildfly (Applicable only for DRONE 2.2.x Versions)

This section is only applicable to Wildfly distribution of Keycloak.

Configuring port and hostname

The script we will use by default is configured with the following file KEYCLOAK_INSTALL_DIR/standalone/configuration/standalone.xml

```
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="${jboss.socket.binding.port-offset:0}">
        <socket-binding name="management-http" interface="management" port="${jboss.management.http.port:9990}"/>
        <socket-binding name="management-https" interface="management" port="${jboss.management.https.port:9993}"/>
        <socket-binding name="ajp" port="${jboss.ajp.port:8009}"/>
        <socket-binding name="http" port="${jboss.http.port:8080}"/>
        <socket-binding name="https" port="${jboss.https.port:8443}"/>
        <socket-binding name="txn-recovery-environment" port="4712"/>
        <socket-binding name="txn-status-manager" port="4713"/>
        <outbound-socket-binding name="mail-smtp">
            <remote-destination host="localhost" port="25"/>
        </outbound-socket-binding>
    </socket-binding-group>
```

By default, Keycloak will use **8080** as an http port, **8443** as an https port and **localhost** as host. If you decided to use custom **KEYCLOAK_HOST_NAME,** it should be specified as shown in bold below.

```
<server name="default-server">
    <http-listener name="default" socket-binding="http" redirect-socket="https" enable-http2="true"/>
    <https-listener name="https" socket-binding="https" security-realm="ApplicationRealm" enable-http2="true"/>
    <host name="default-host" alias="keycloak.mycomp.any">
        <location name="/" handler="welcome-content"/>
        <http-invoker security-realm="ApplicationRealm"/>
    </host>
</server>
```

Initial Startup of Keycloak Server

Run the standalone version of the server in **KEYCLOAK_INSTALL_DIR/bin**

For Linux

```
standalone.sh
```

For Windows

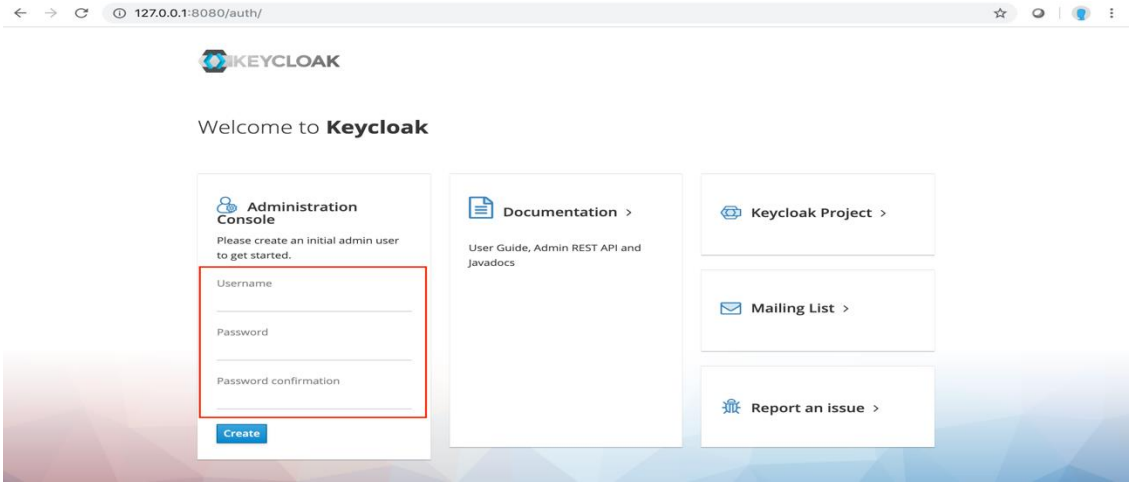
```
standalone.bat
```

Creating a Realm Administrator

Note: This step must be performed on a machine running Keycloak.

Open a browser and navigate to https://KEYCLOAK_HOST_NAME:8443/auth (or HTTP endpoint).

Note: the browser may show a warning if e.g. certificate does not match the host name



Specify the admin credentials and press Create.

This admin user is a super user with all full access to Keycloak (realm creation, update, deletion, user creation, update, deletion, etc.)

Starting Keycloak Server

Stop the Keycloak server and start it again with -b parameter.

Run the standalone version of the server in **KEYCLOAK_INSTALL_DIR/bin**

For Linux

```
standalone.sh -b KEYCLOAK_IP_ADDRESS
```

For Windows

```
standalone.bat -b KEYCLOAK_IP_ADDRESS
```

It is possible to use 0.0.0.0 instead of **KEYCLOAK_IP_ADDRESS** to allow connection from any interface, by default only connections from the same machine will be accepted. See Keycloak documentation corresponding to the version specified in system_requirements.pdf for details.

Importing Previous Configuration

If you configured drone realm in version earlier than 2.3 of Design Room ONE you need to perform the following steps to keep your configuration, i.e., users and roles. If you are doing a fresh installation for Design Room ONE proceed with the next section.

1. Export your current realm data
    c. Ensure your keycloak server instance is stopped
    d. Run the below command to export your current drone Keycloak realm data

    Run the command below:

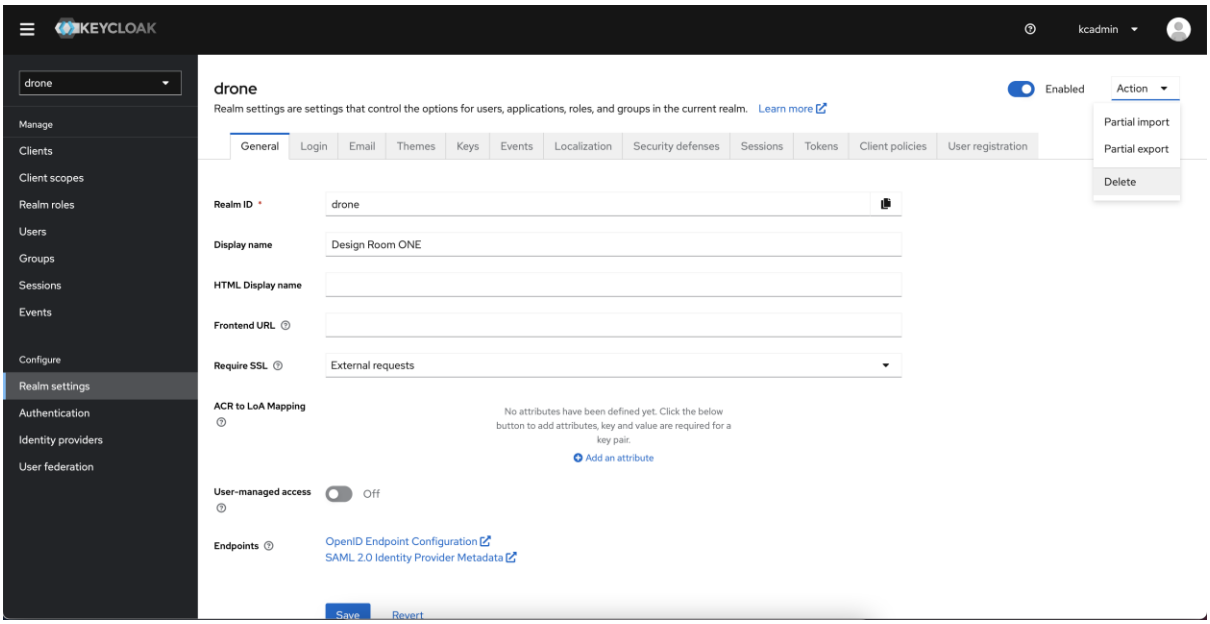
For Linux

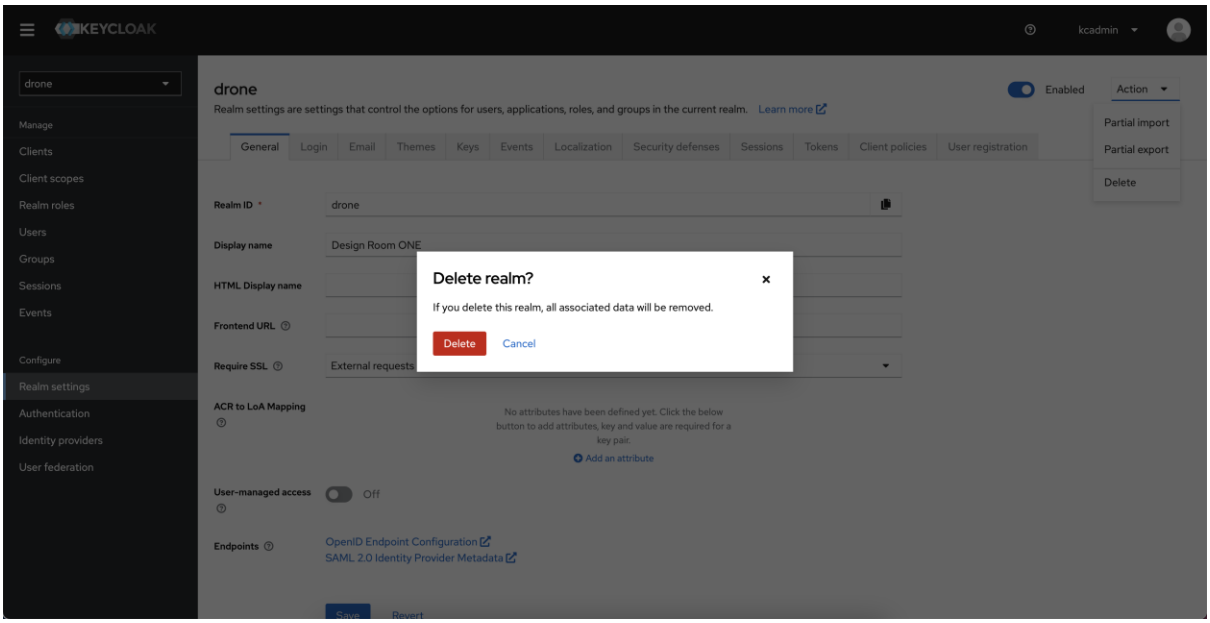
```
KEYCLOAK_INSTALL_DIR/bin/standalone.sh -
 Dkeycloak.migration.action=export -
 Dkeycloak.migration.provider=singleFile
 -Dkeycloak.migration.file=/tmp/myOldRealm.json
```

For Windows

```
KEYCLOAK_INSTALL_DIR\bin\standalone.bat -
 Dkeycloak.migration.action=export -
 Dkeycloak.migration.provider=singleFile
 -Dkeycloak.migration.file={path_to_temp_folder}\myOldRealm.json
```

2. Delete existing drone realm
    a. Start your Keycloak server instance
    b. Choose the **Realm** (i.e. drone) in the dropdown on the left panel.
    c. Navigate to “Realm Setting” and then click on **Actions** -> **Delete** to DRONE realm as shown below.





3. Import Design Room ONE provided sample realm

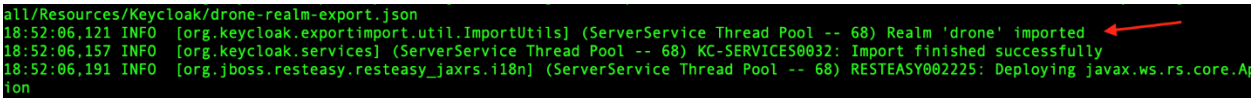
   Run the following command

   For Linux

   ```
   KEYCLOAK_INSTALL_DIR/bin/standalone.sh -
   Dkeycloak.migration.action=import -
   Dkeycloak.migration.provider=singleFile -Dkeycloak.migration.file=
   DR_ONE_INSTALL_DIR/DR_Install/Resources/Keycloak/drone-realm-
   export.json
   ```

   For Windows

   ```
   KEYCLOAK_INSTALL_DIR\bin\standalone.bat -
   Dkeycloak.migration.action=import -
   Dkeycloak.migration.provider=singleFile -Dkeycloak.migration.file=
   DR_ONE_INSTALL_DIR\DR_Install\Resources\Keycloak\drone-realm-
   export.json
   ```



   The Keycloak server will start, you should stop it again.

4. Import your exported realm

   Run the command below:

   For Linux

   ```
   KEYCLOAK_INSTALL_DIR/bin/standalone.sh -
   Dkeycloak.migration.action=import -
   Dkeycloak.migration.provider=singleFile -
   Dkeycloak.migration.file=/tmp/myOldRealm.json
   ```

   For Windows

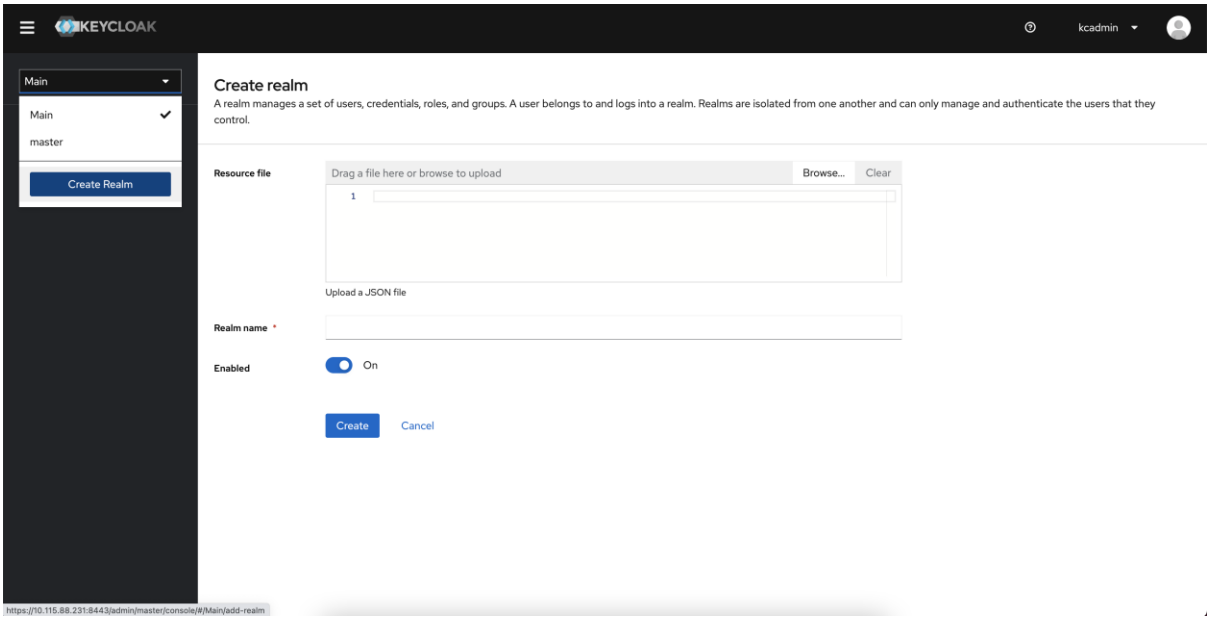
```
KEYCLOAK_INSTALL_DIR\bin\standalone.bat -
 Dkeycloak.migration.action=import -
 Dkeycloak.migration.provider=singleFile -Dkeycloak.migration.file=
 {path_to_temp_folder}\myOldRealm.json
```

5. Start the Keycloak server, see this section for details.
6. Ensure that all previously created users have **view-realm** role as explained in creating users section
7. Realm configuration is now complete, you should skip instructions in the next section and proceed with Setting up Users and Roles.

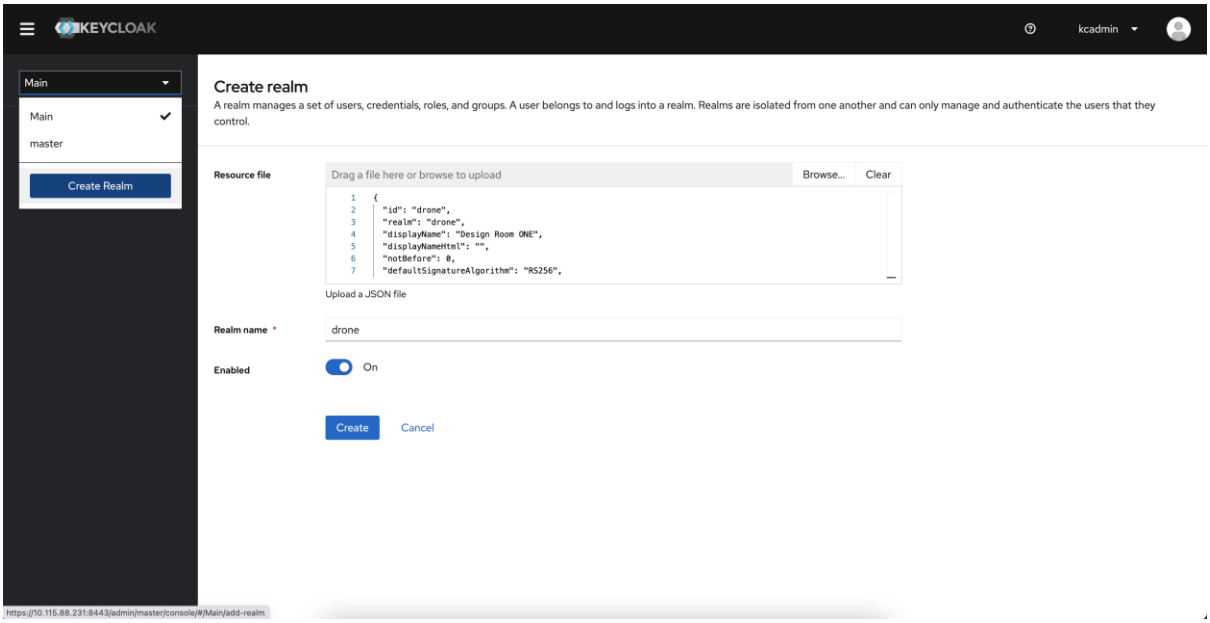
Configuring the Realm

Note: this step is easier to do from the machine where Design Room ONE is installed

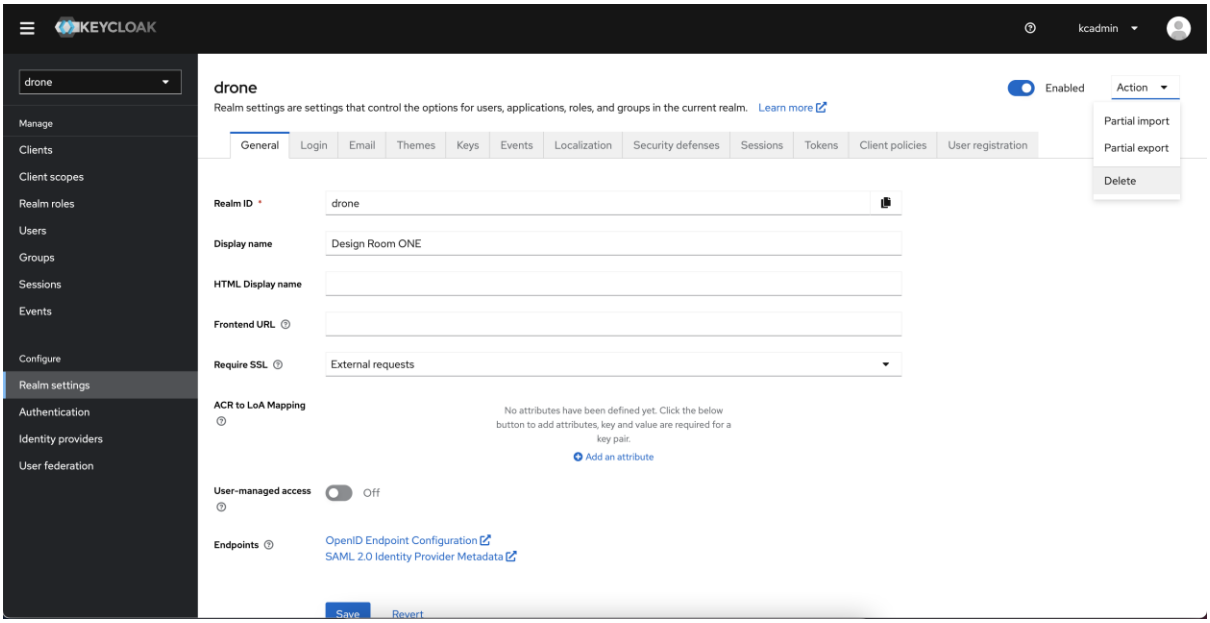
1. Import the realm data
    a. Hover over **Realm** dropdown in top left corner and click on **Create Realm** and wait for it load to the below page.



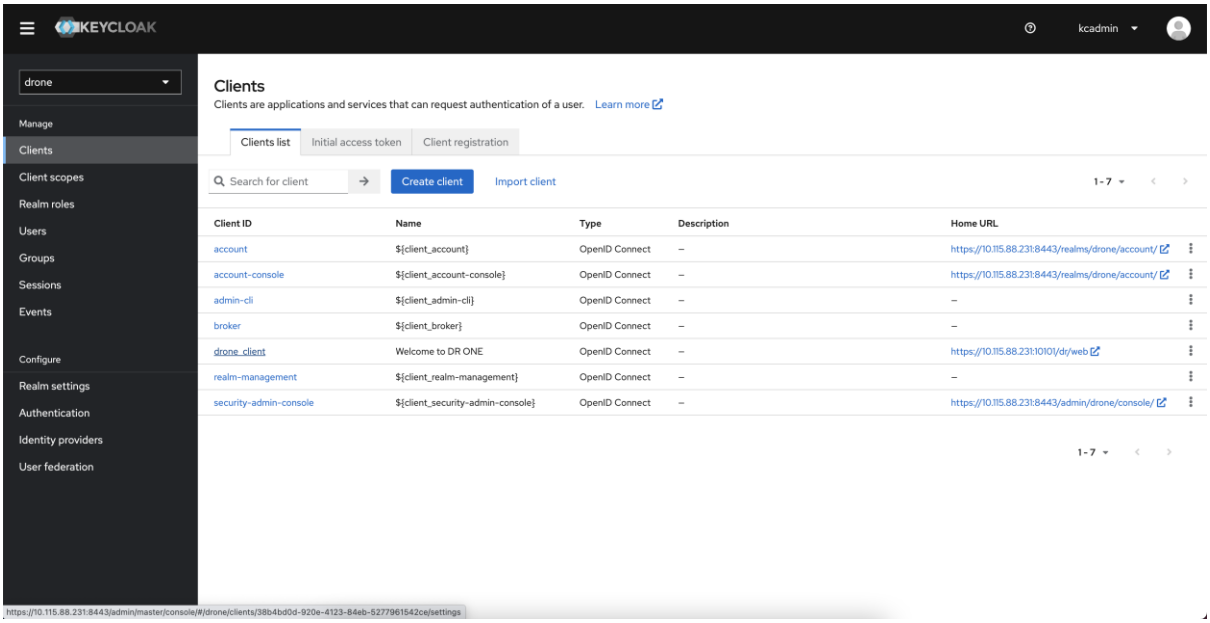
    b. Click on **Select File** and point to **drone-realm-export.json** file. It can be found in machine where **Design Room ONE** is installed under following path: DR_ONE_INSTALL_DIR/DR_Install/Resources/Keycloak/drone-realm-export.json
    c. Then, click **Create**



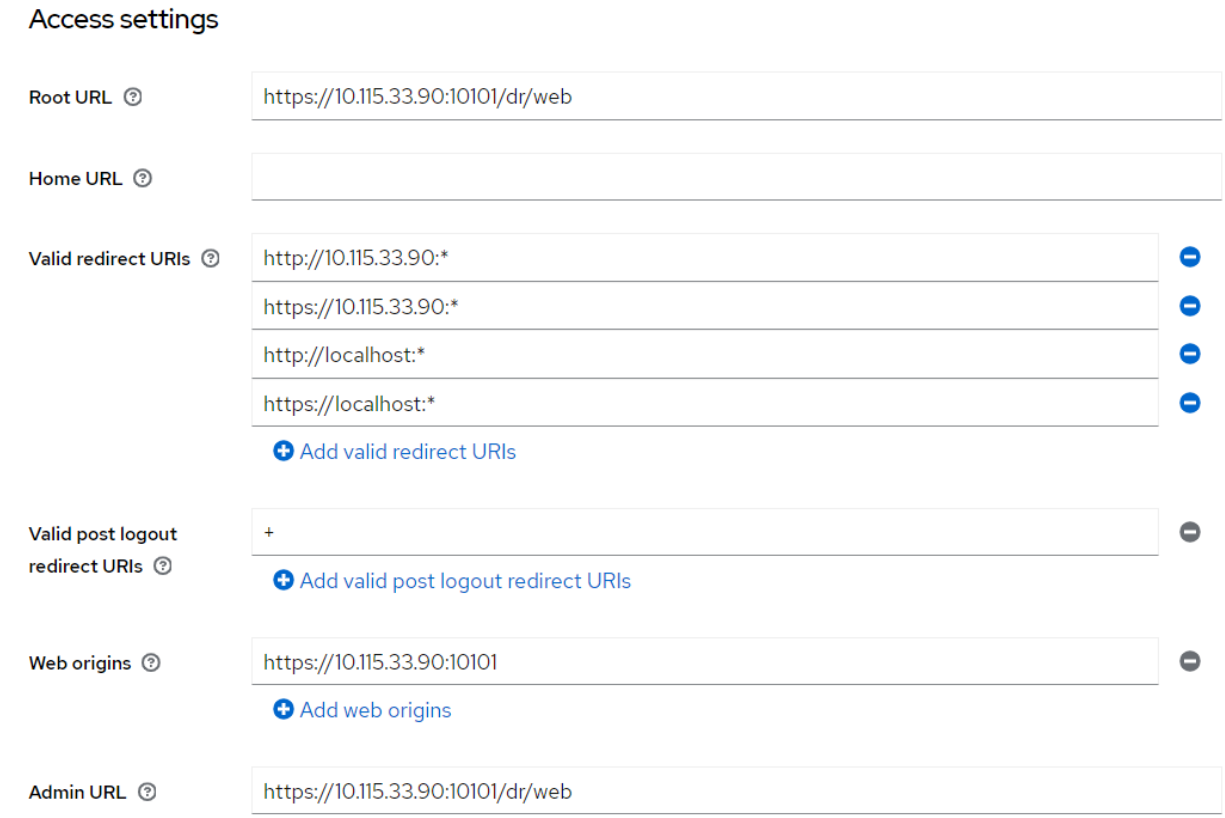
d. Your **Drone** realm should be created successfully as shown below:



2. Setup drone_client
    a. Under **Clients** click on the drone_client link

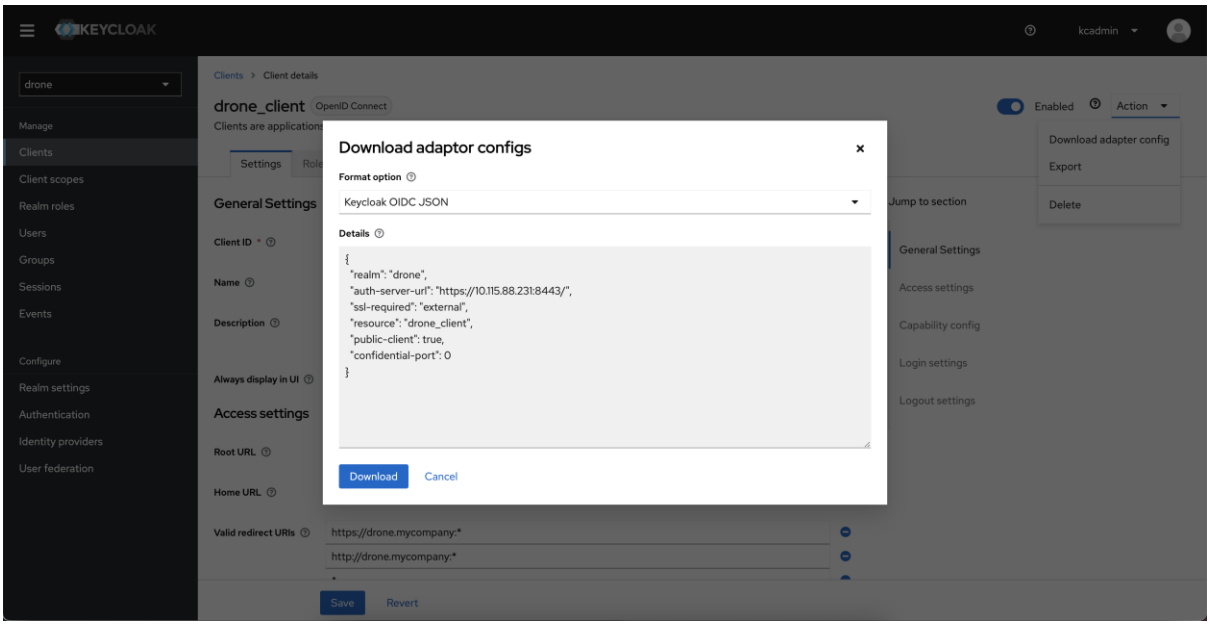


b. Click on the **Settings** top menu, scroll down and add **DRONE_HOST_NAME** (or ip address) accessible by other machines for the server Keycloak as valid redirect URIs as shown in example below. It is recommended to use lowercase letters.



c. Make sure **Root URL**, **Admin URL**, and **Web Origins** fields are also updated with **DRONE_HOST_NAME** and hit **Save**

d. On the top right **Action** dropdown, select **Download adaptor config** menu.
e. In the **Download adaptor configs** popup, select **Keycloak OIDC JSON** in **format option** and click **Download**.



Note : Copy and paste **auth-server-url** and **ssl-required** properties in the file **DR_ONE_INSTALL_DIR/OnPrem_Design_Room/config/server-config.json** under **dr_keycloak_config** object. Also change **"dr_auth"** attribute value to **"keycloak"**. Then the configuration file would look like this.

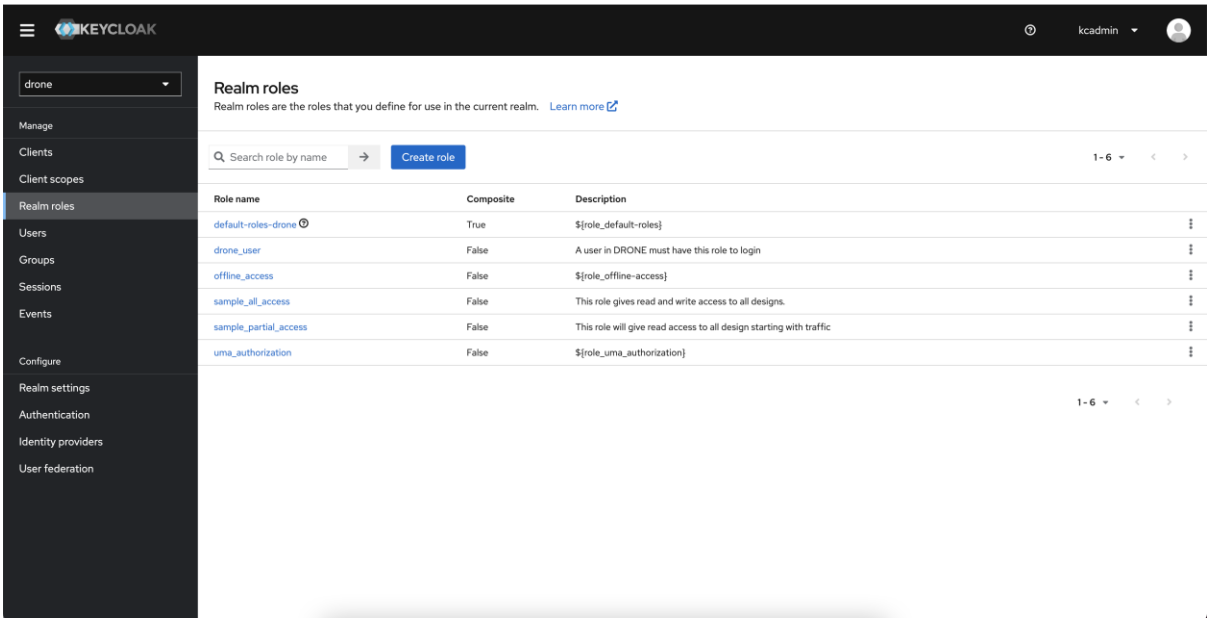
```
  // Authentication (for accessing information stored in Design Room ONE)
  // "none": Do not use any authentication. Everyone can access all designs.
  // "jazz": Use Jazz authentication. User needs to be logged in to Jazz to acces
s designs.
  // "keycloak": Use Keycloak authentication. User needs to be logged via keycloa
k to access designs.
  "dr_auth": "keycloak",

  //Keycloak configuration
  "dr_keycloak_config": {
    "auth-server-url": "https://keycloak.mycomp.any:8443/auth/",
    //Defines security level for Keycloak server
    //"none": HTTPS not required for any IP address
    //"external": Private IP and localhost can access without HTTPS
    //"all": HTTPS required for all IP addresses
    "ssl-required": "external"
  }
```

Note: Make sure correct **KEYCLOAK_HOST_NAME** (or ip-address) specified in **auth-server-url** attribute and the URL is accessible by machine where Design Room ONE server is installed and from user machines. It is recommended to use lowercase letters in the URLs.

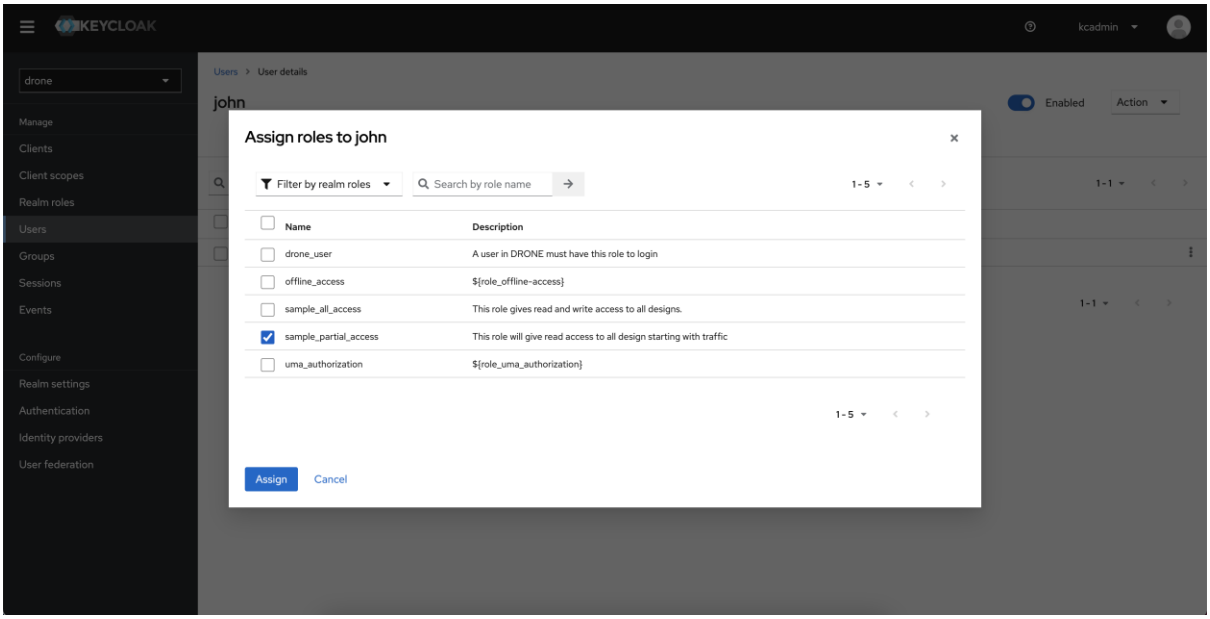
Setting up Users and Roles

Creating Users

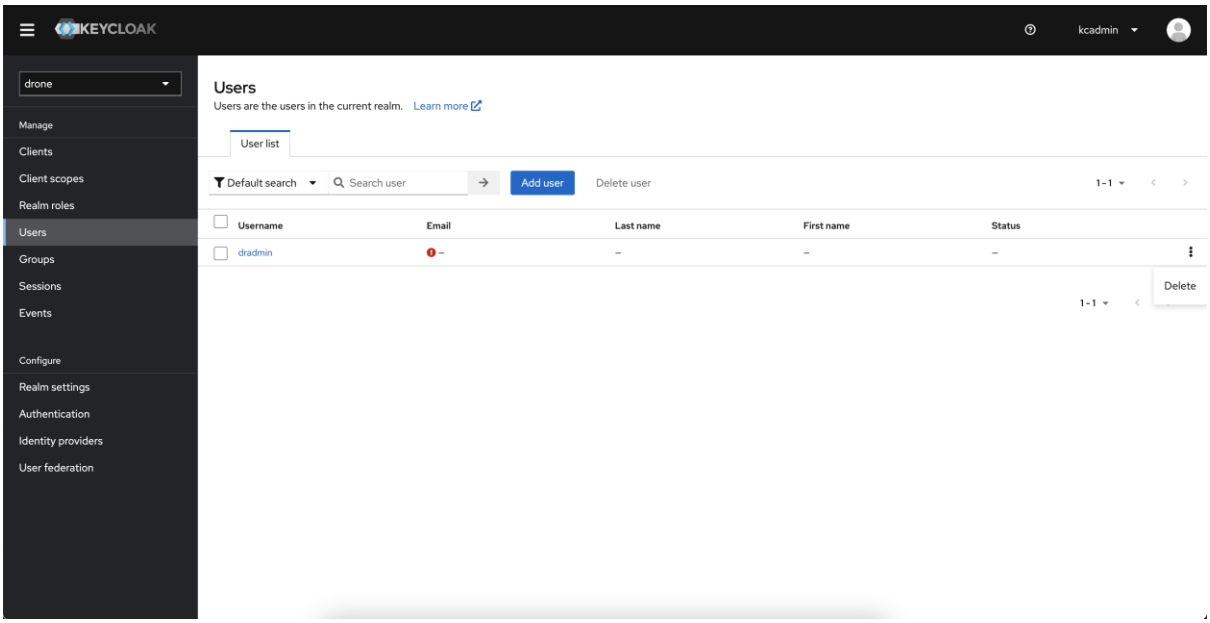
Before we create a user, you can notice that some roles were created by default in Keycloak as below.



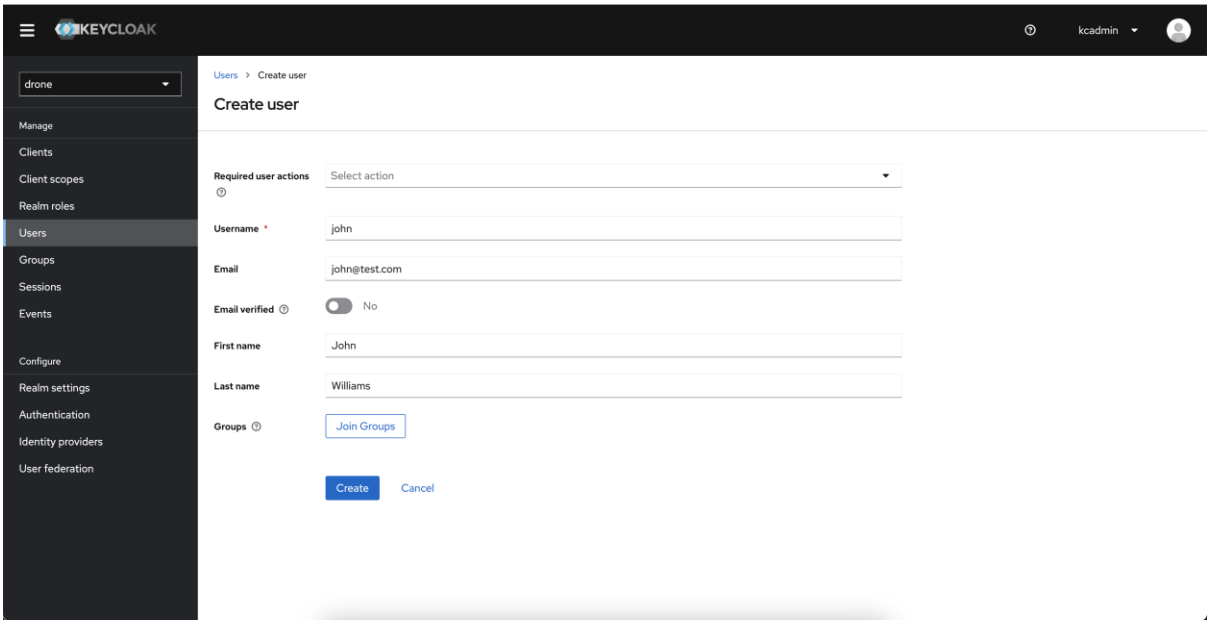
Note: Under the **Realm Roles** tab, as shown below, you will notice that **drone_user** is a default role and is automatically assigned to newly created users. These roles are **required** by the Design Room ONE server normal operation.



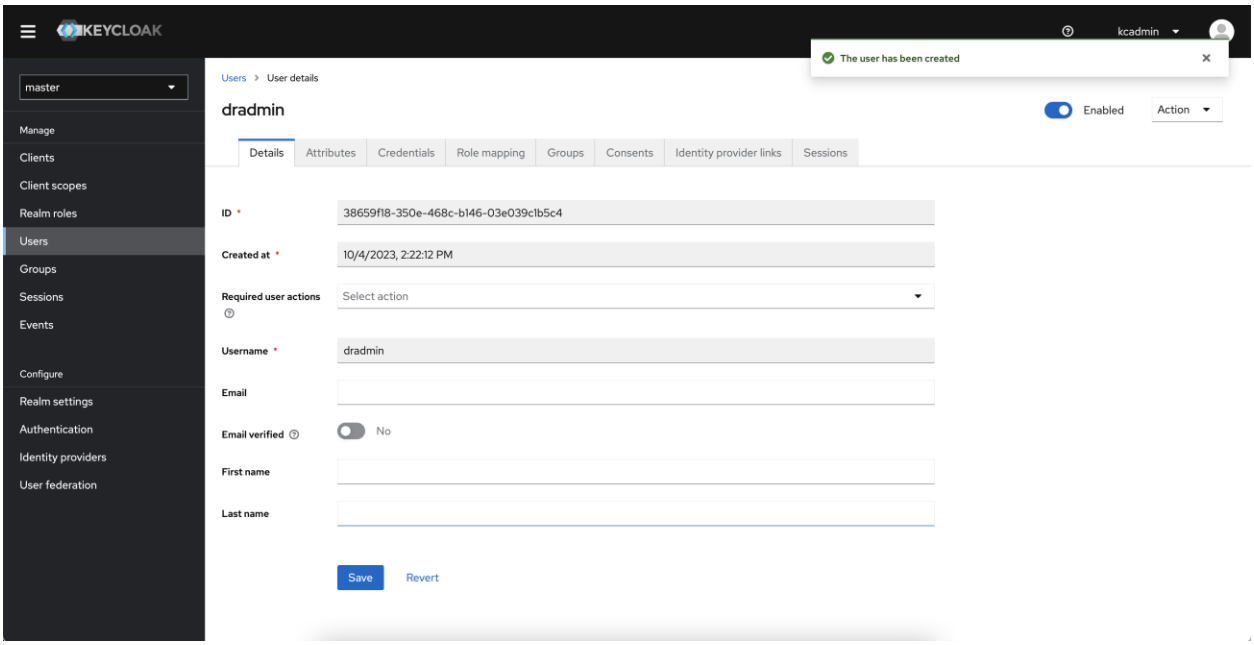
1. Create a user by clicking **Add user**.



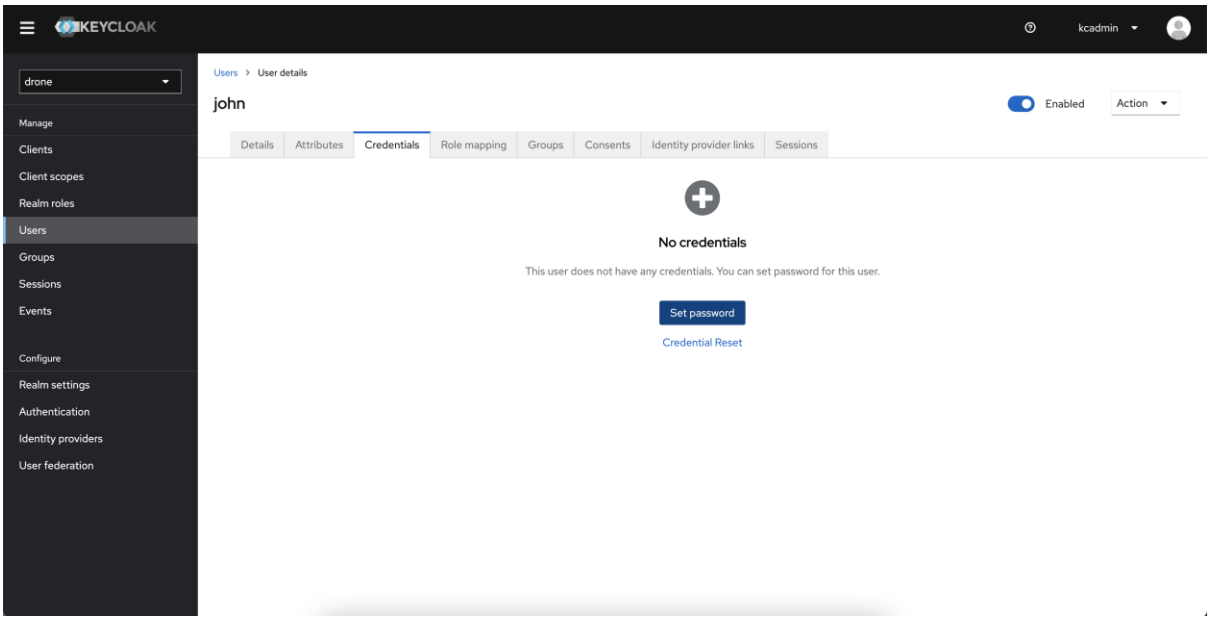
2. Fill in the necessary user information and click on **Create**.



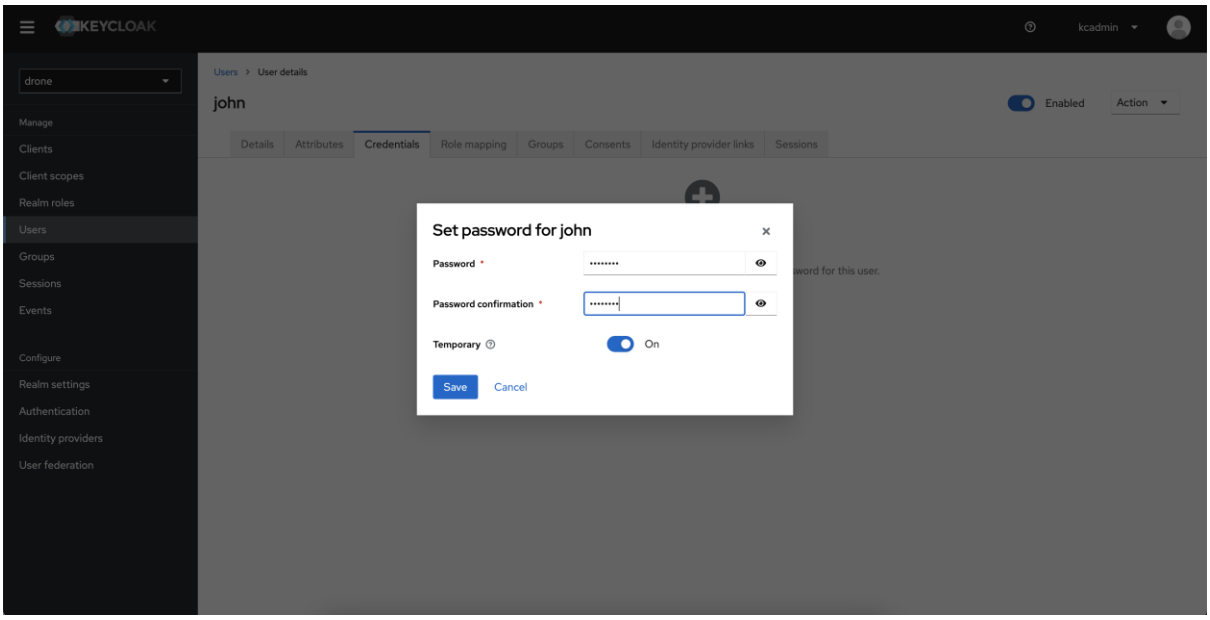
3. Verify the details in **User details** page.



4. Switch to **Credentials** tab and click on **Set password.**



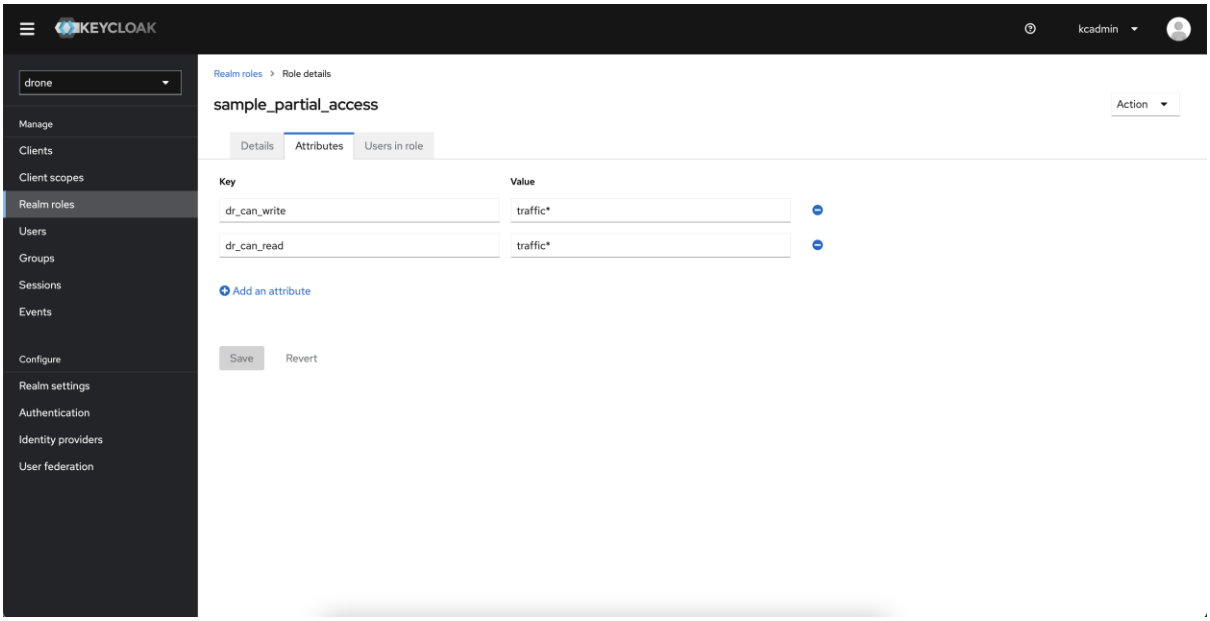
5. Enter the **Password** details, select **Temporary** preference toggle and click on **Save**.



Managing Access with Roles

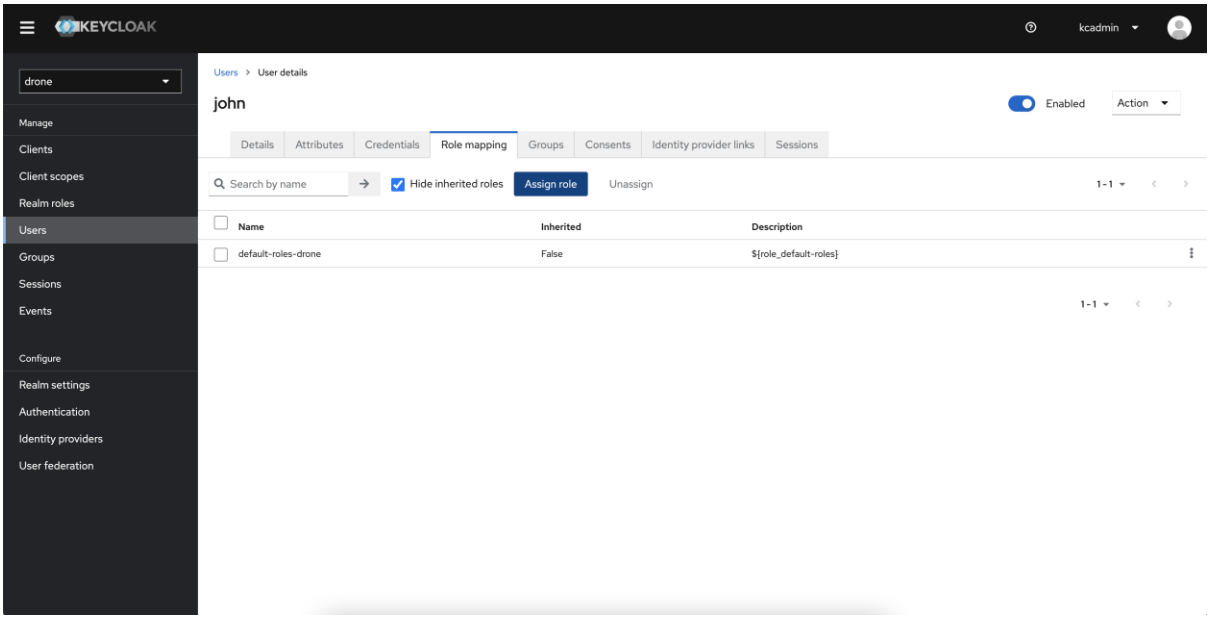
Access to designs in Design Room ONE is controlled with special attributes that can be specified for a role in Keycloak. The **dr_can_read** and **dr_can_write** attributes of roles give users respectively read and write access to specific designs. These attributes support wildcard as shown in the picture below.

1. Under **Realm Roles > sample_partial_access > Attributes,** you can see that the **dr_can_read** and **dr_can_write** attributes are both set to “traffic*” which means that users or groups with this role will be able to read and write to all designs with names starting with “traffic”.

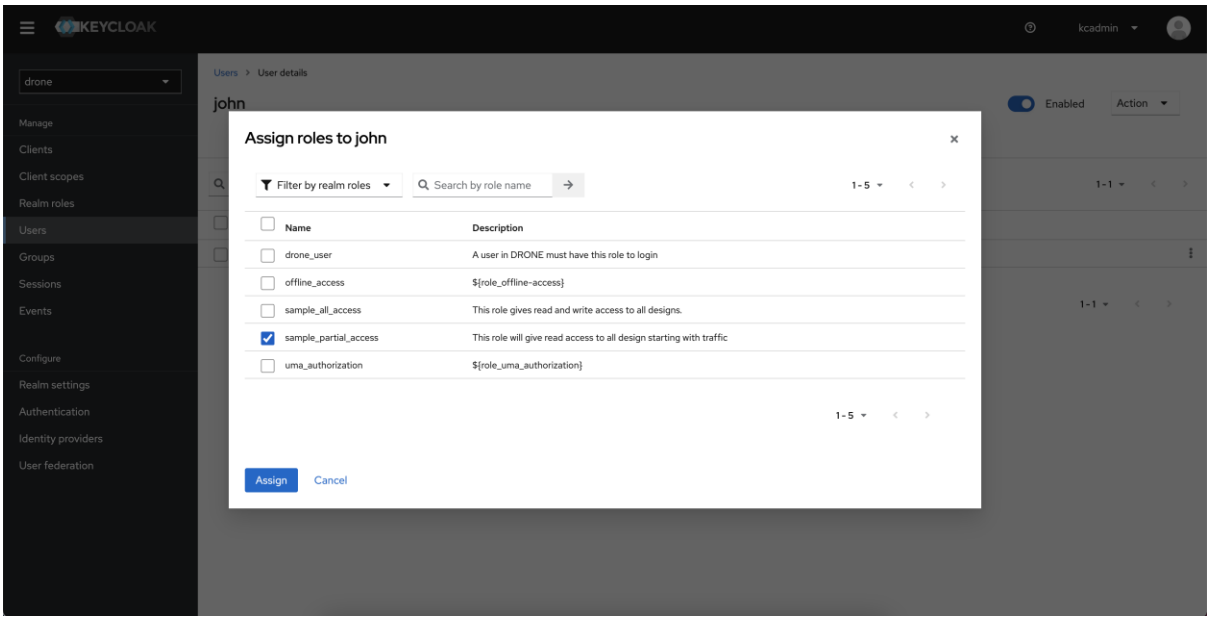


Note: It is possible to specify several alternatives by using pipe (|) character.e.g., the following value **traffic*|*_secretproject|*alice*** will give read or write access (depending on the attribute it is specified in) to designs, which name either starts with **traffic**, or ends with **_secretproject** or contains the word **alice**. Note that design names are case sensitive.

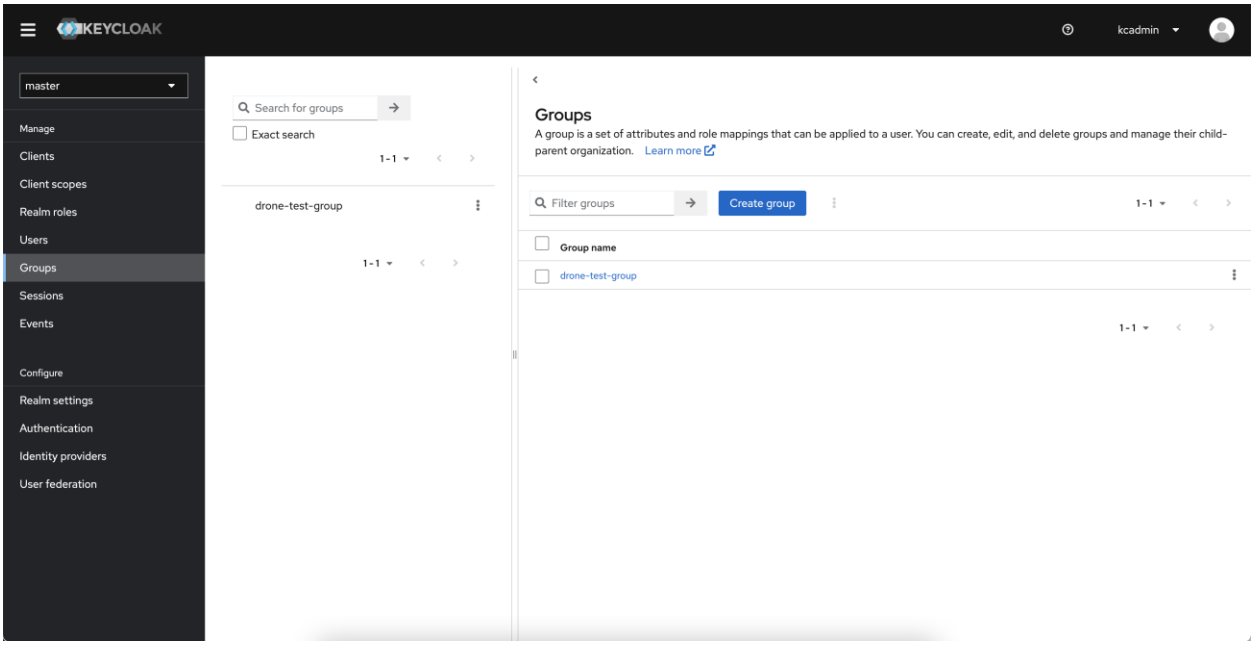
2. To explicitly assign a role to a user, **Users** > Select the user > **Role Mapping**, click on Assign role.



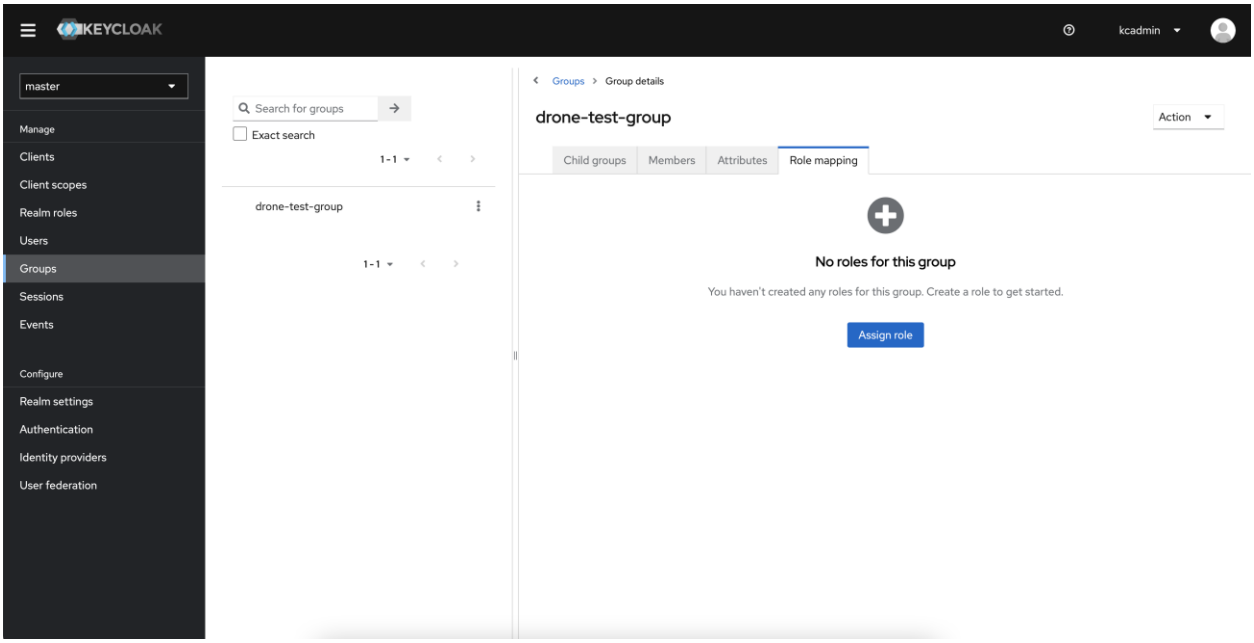
3. Select the roles to be assigned for the user and click **Assign.**



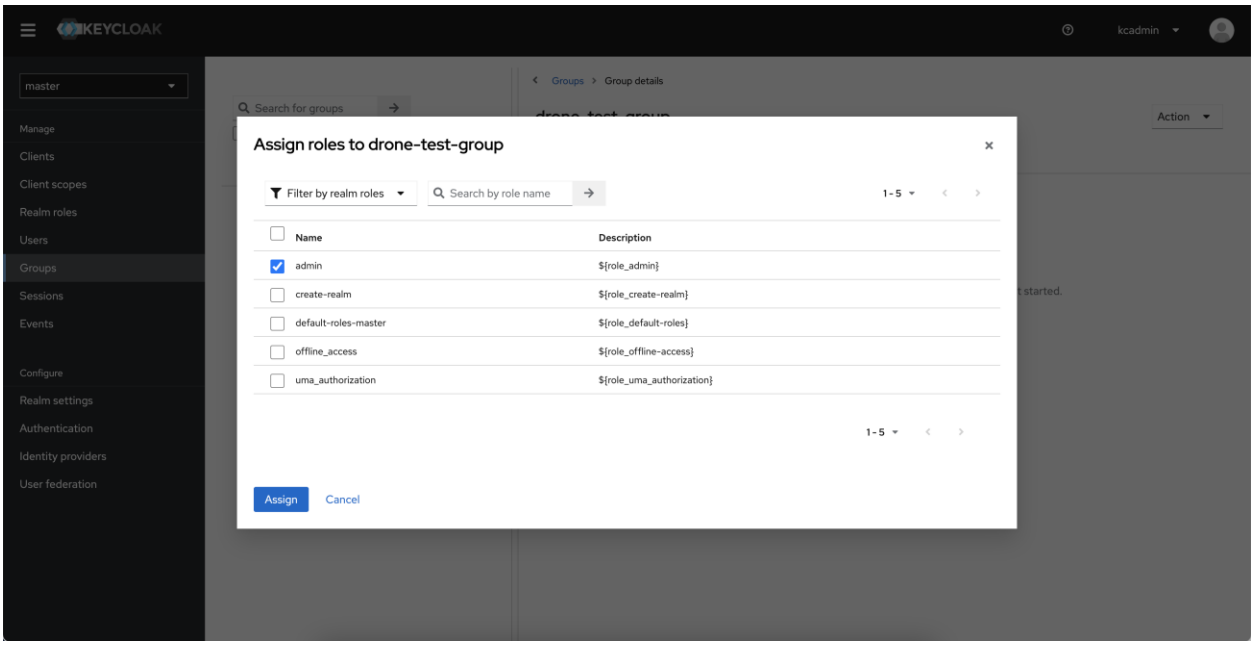
4. A user can implicitly get roles from the groups they belong to. To assign a role to a group, navigate to **Groups** > Select the group.



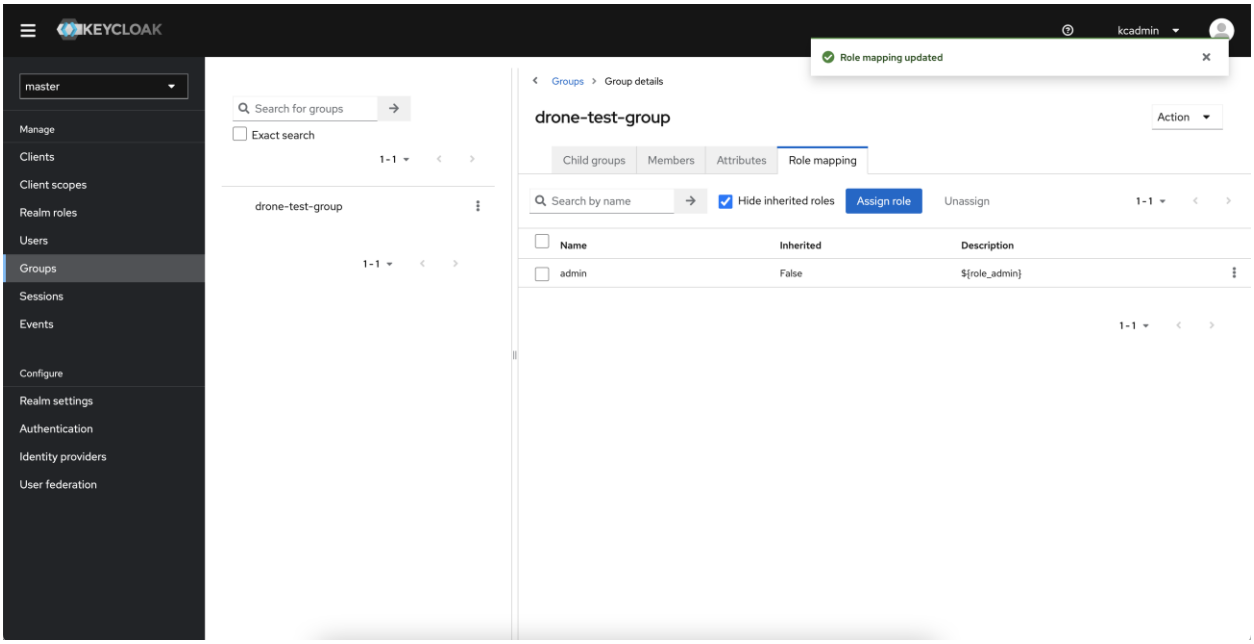
5. Under Role Mapping, click on **Assign role.**

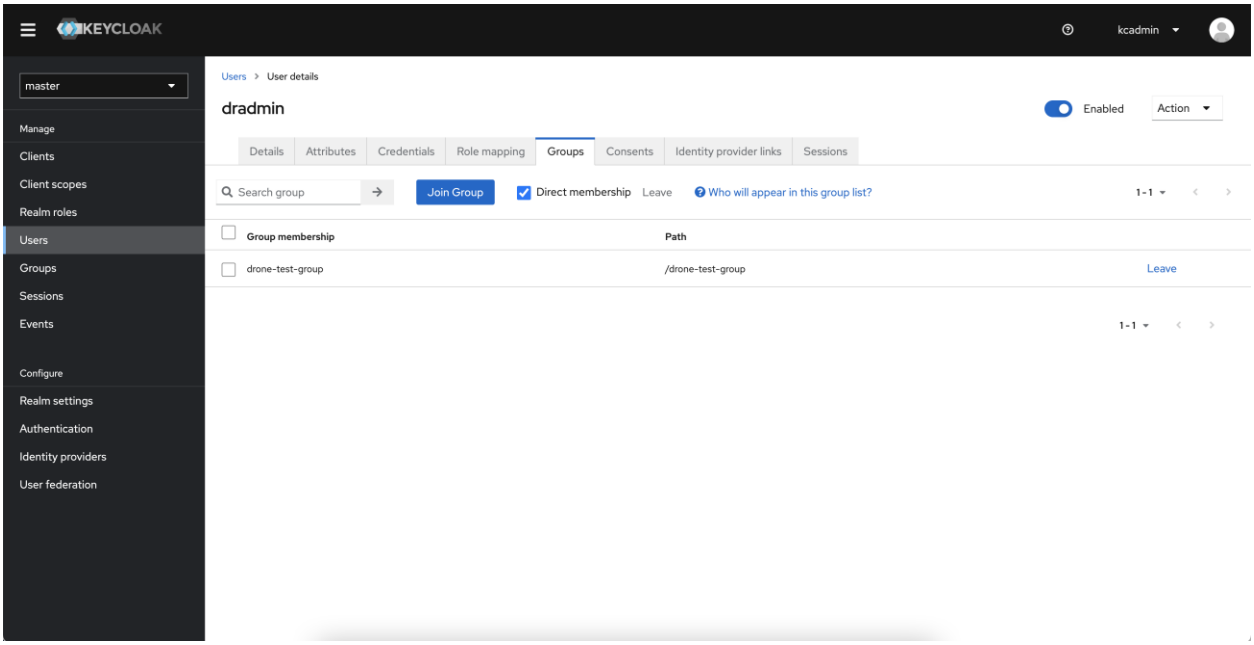


6. Select the role to be assigned and click on **Assign**.



7. Verify the changes below.





Note:

a) A user will inherit all the roles from the groups they belong to. In the example below user **dradmin** belongs to **drone-test-group.**

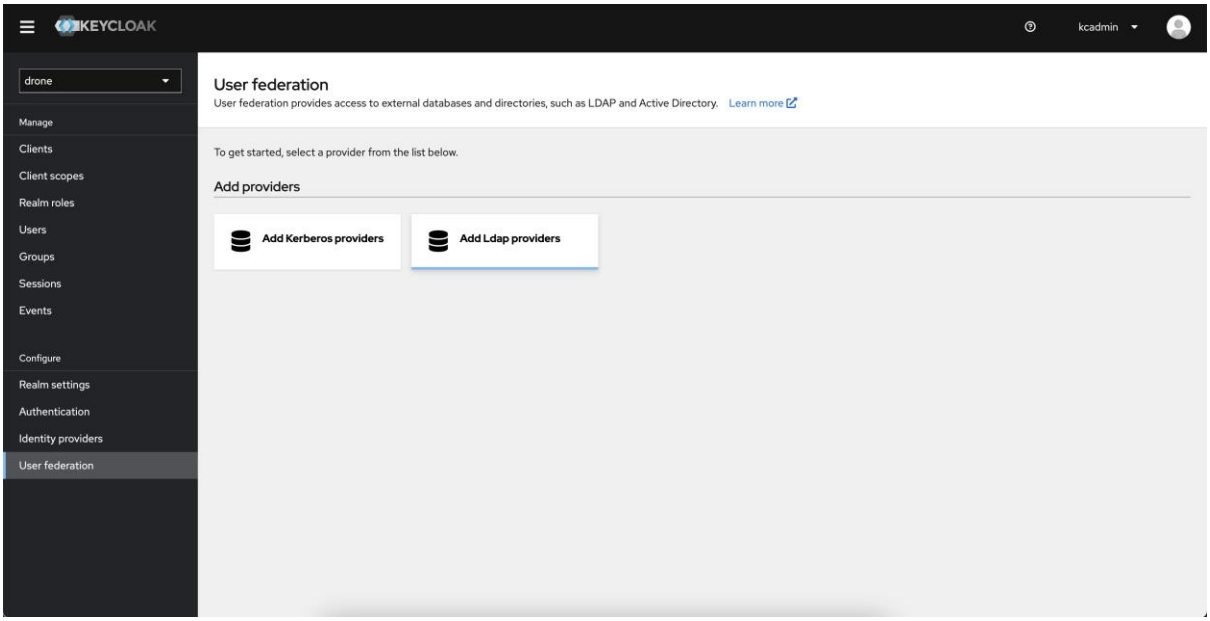
b) This means user **dradmin** automatically inherits **admin** role because it is assigned to the group.

c) This results in any **my_partial_access_group** member including user **Steven** being able to read and write all designs with names starting with “traffic” because of the group membership.

Importing Users from Active Directory

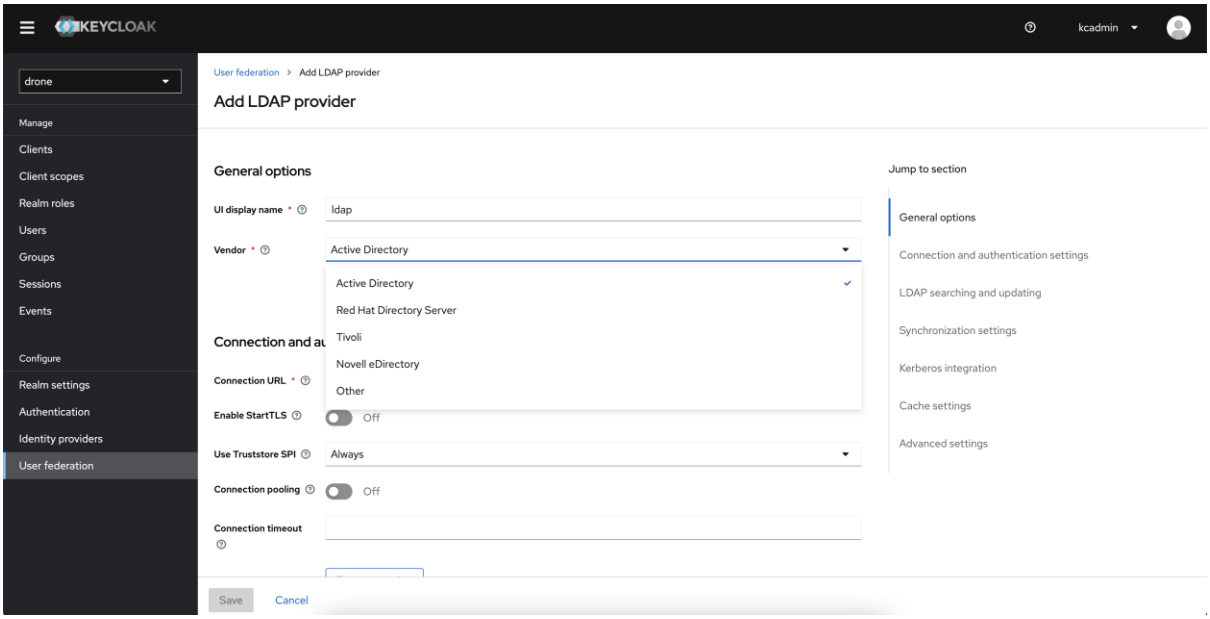
Note: The Active Directory server reference is at the end of this section

1. Ensure you are in the **Drone** realm and click on **User Federation** left menu item
   Then select **ldap** from the dropdown

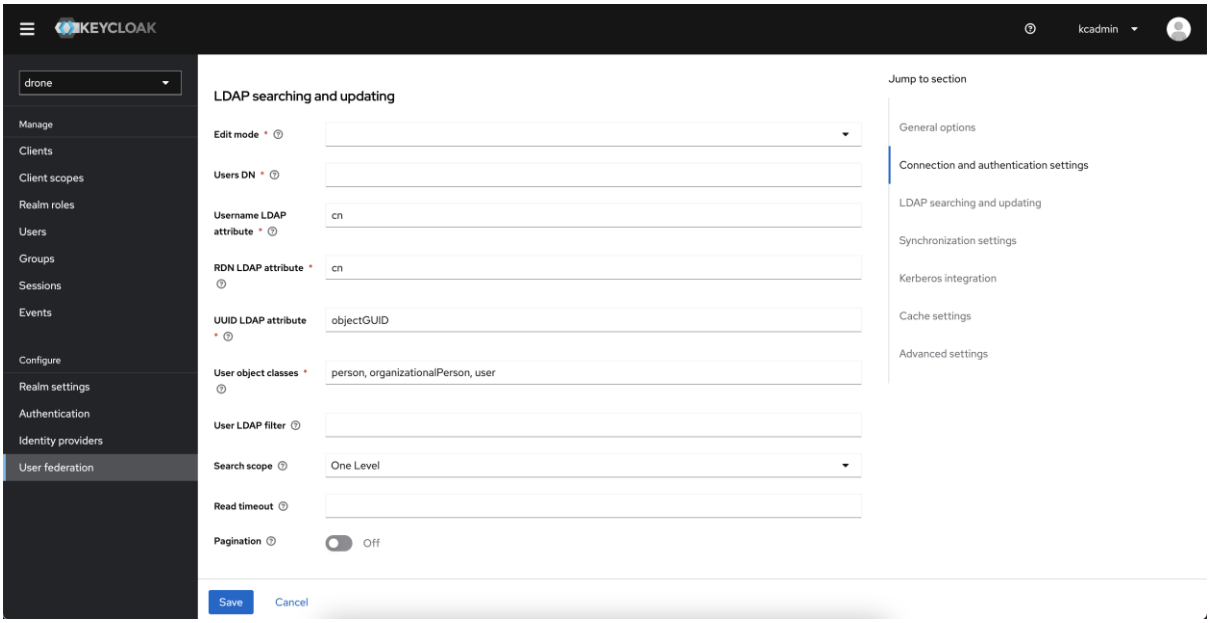


Note: Basic Active Directory configuration will be covered in this section. Additional configuration options are available in Keycloak with context sensitive help.

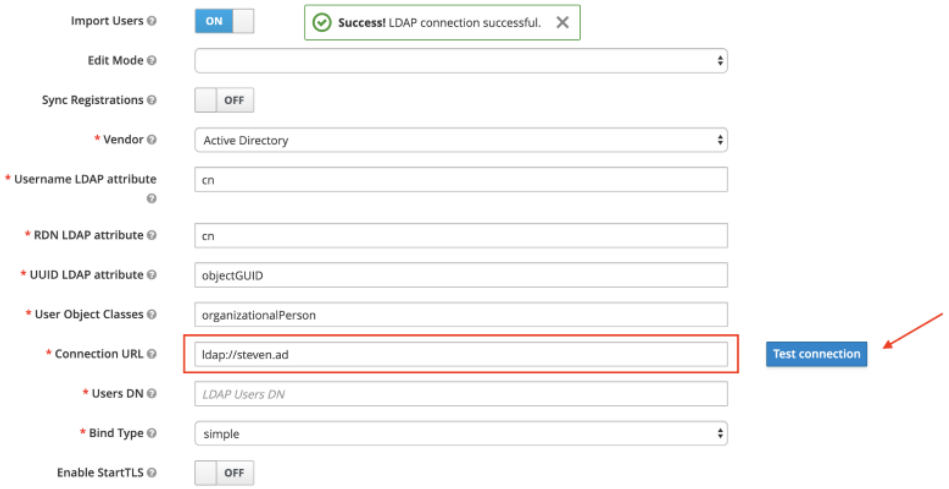
2. Set **Vendor** to **Active Directory**



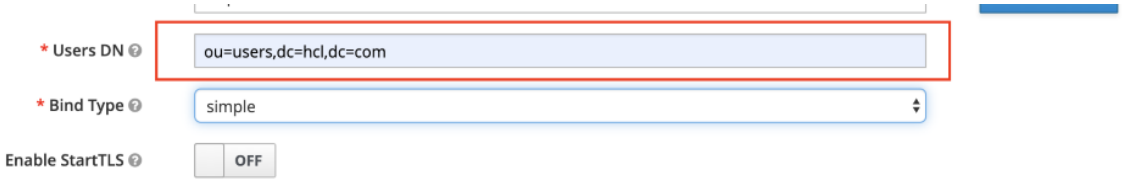
You should see some fields pre-filled with default values as shown below:



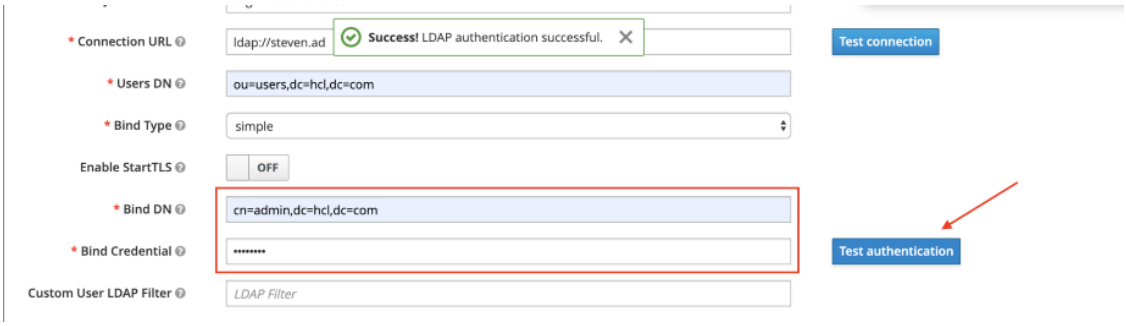
3. Set the connection url and test the connection via the **Test connection** button as shown below:



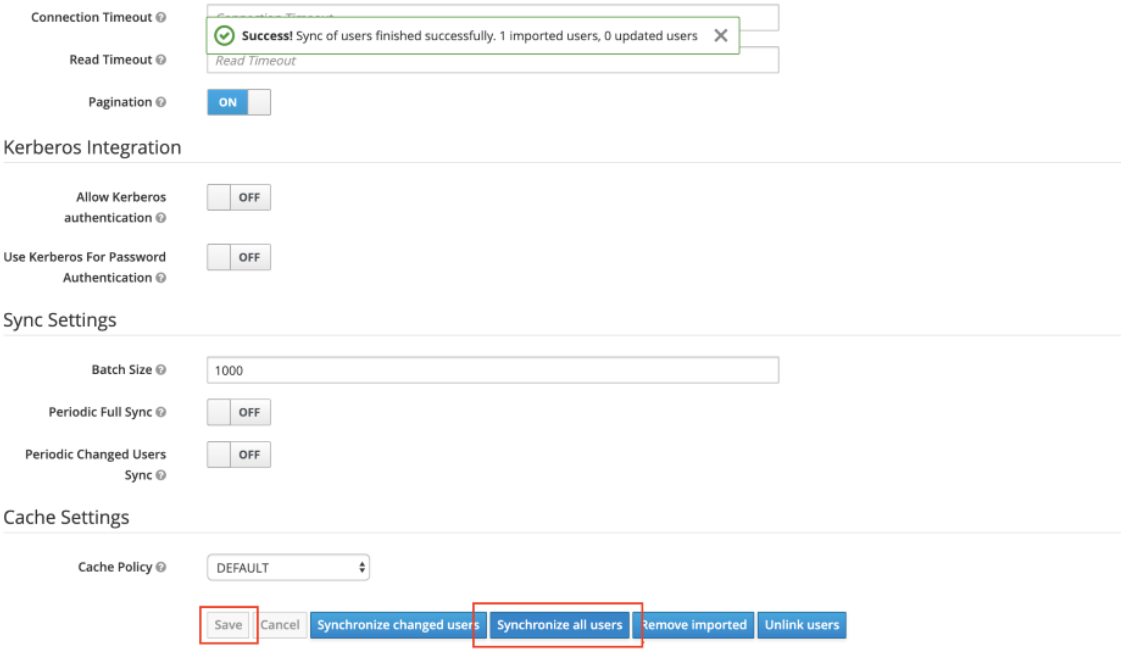
4. Set the active directory users database here.



5. Set the Active Directory (AD) admin credentials and test authentication to the AD server as shown below:

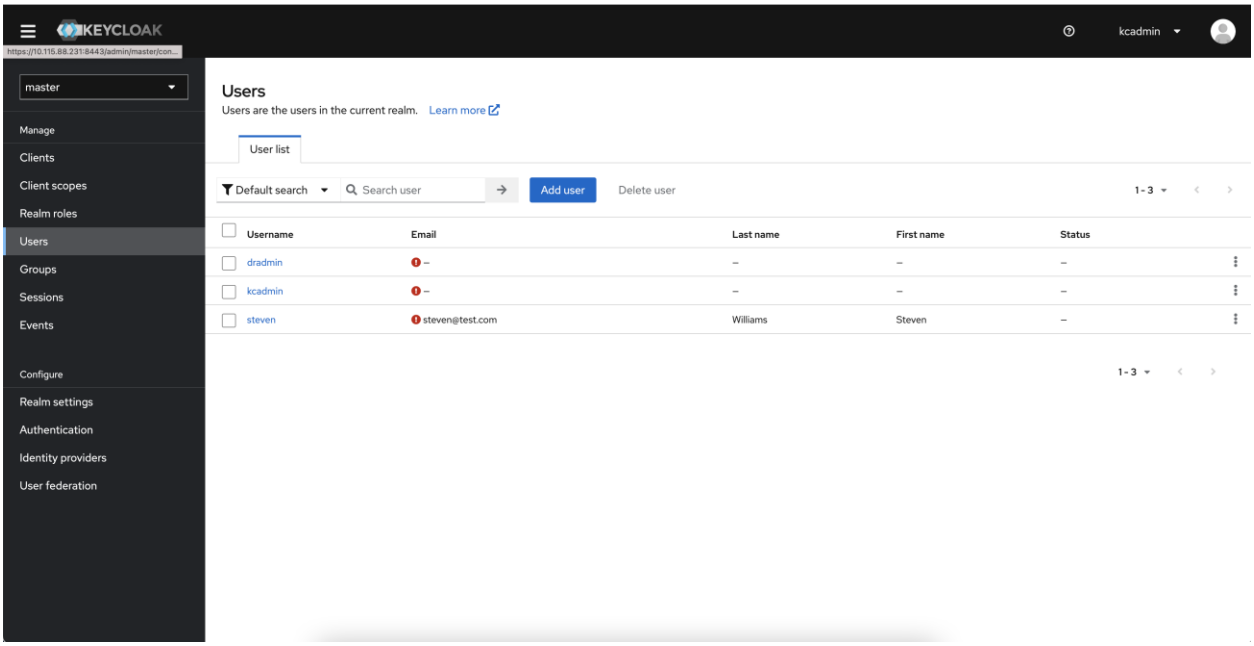


6. Leave the rest as is and scroll down and click **Save**, then click on **Synchronize all users** to import all existing users from AD to Keycloak

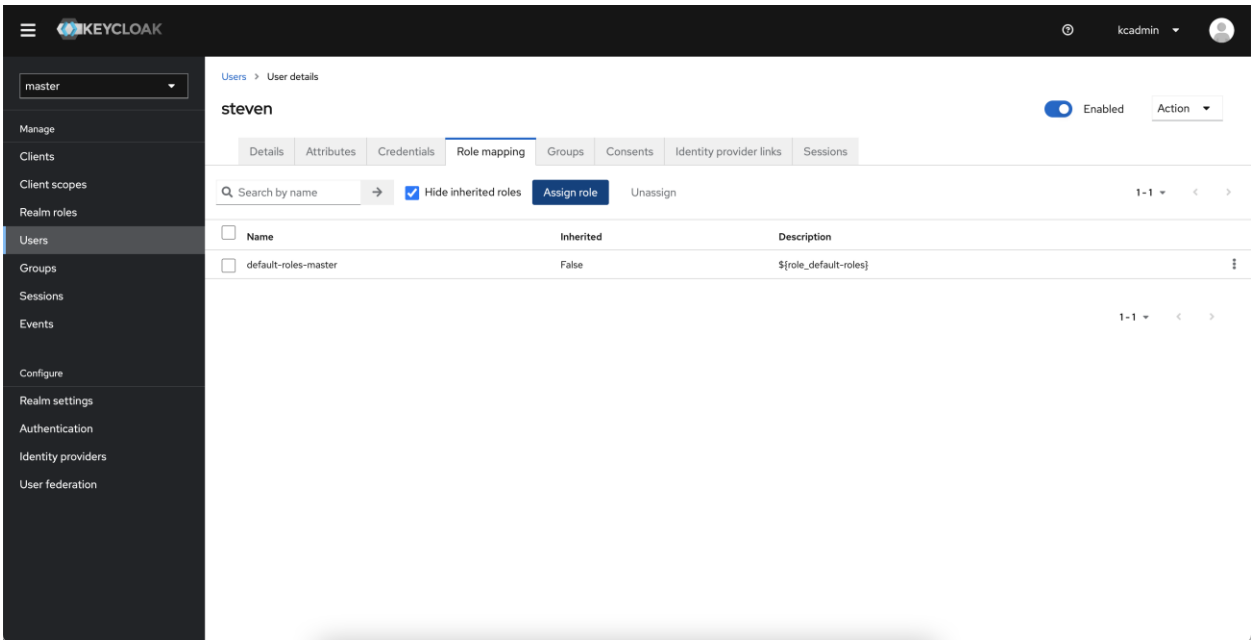


In our case 1, the user was imported from our active directory server.

7. If we now click on the Users menu, we see our imported user.

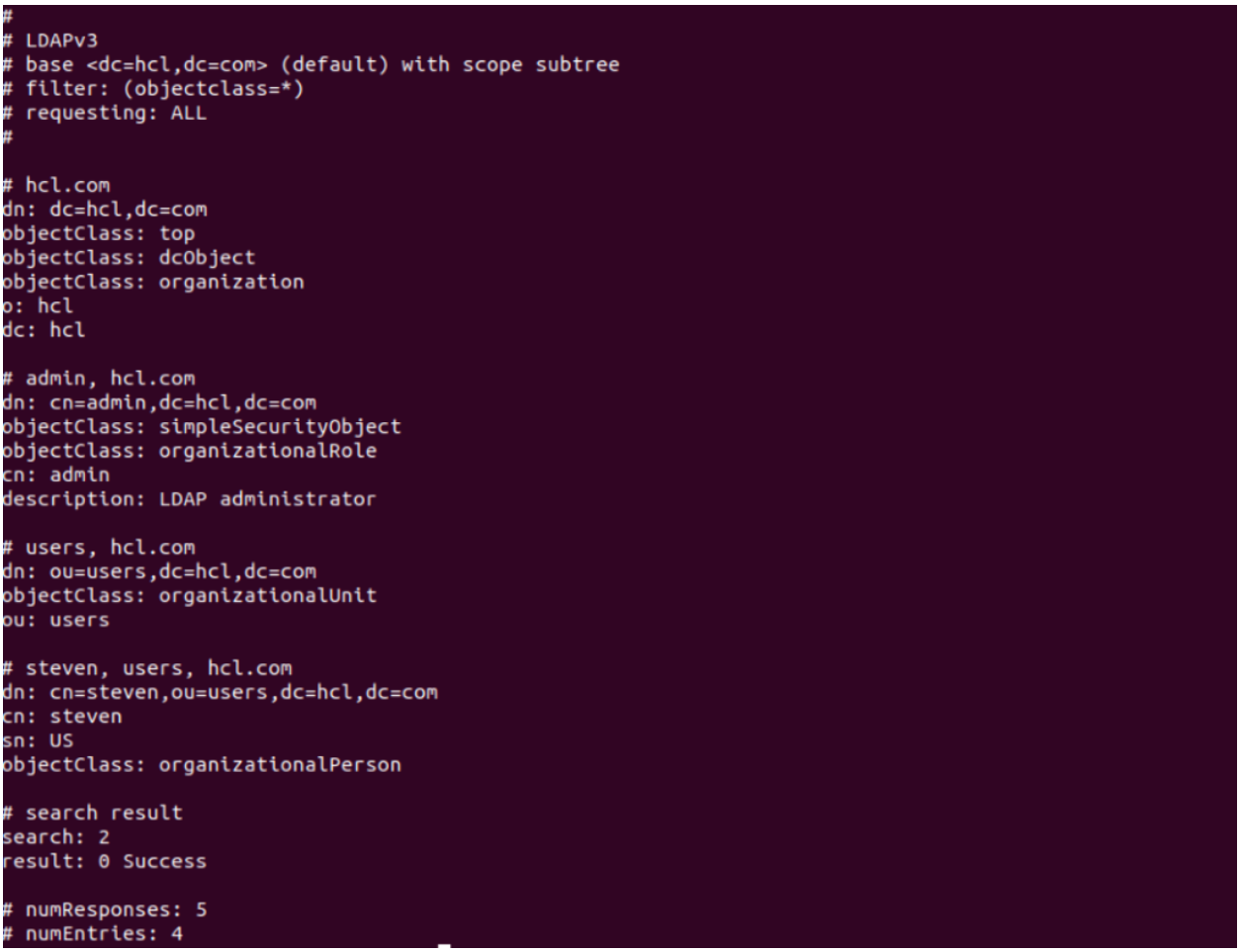


8. Click on the user and observe the role mappings. The user should inherit all default roles (including **drone_user** and **view-realm**) to have access to the Design Room ONE server



9. You can now login into Design Room ONE with the newly created user.

LDAP Directory Information Tree data reference



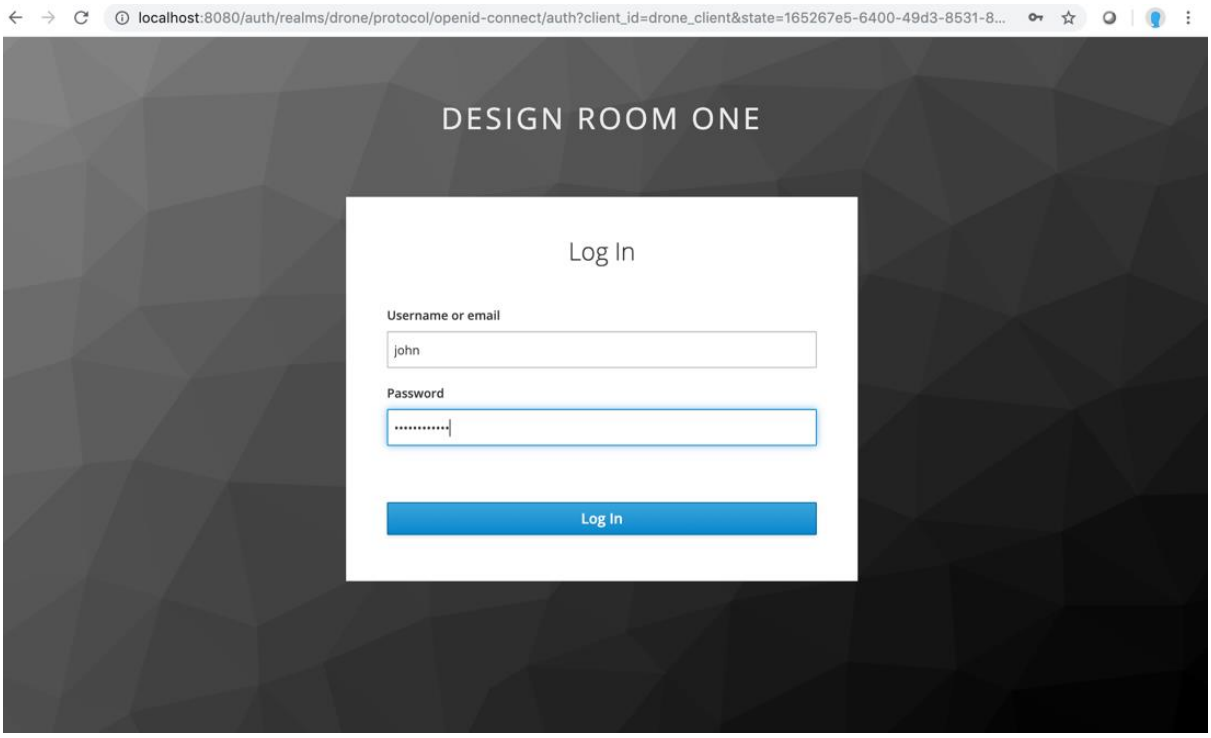
Summary: Hcl.com(organization)->users(organization unit or OU)->steven(Member of OU)

Starting the Design Room ONE Server

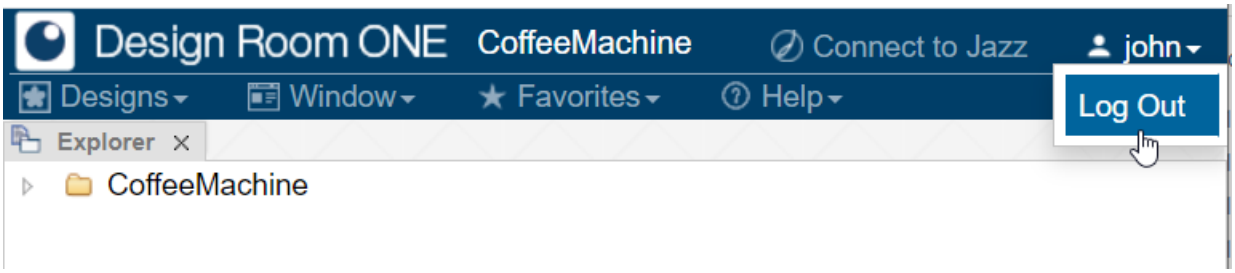
Ensure Design Room ONE server is started. Navigate to your Design Room ONE installation and launch the dr-deploy.js script.

Logging into the Design Room ONE Server with the New User

Users will be redirected to login page when they try to access Design Room ONE resources for the first time



To log out of Design Room ONE click on the username in the top right corner of the page or press SHIFT + L on the keyboard and then click on **Log Out** menu item or hit Enter.



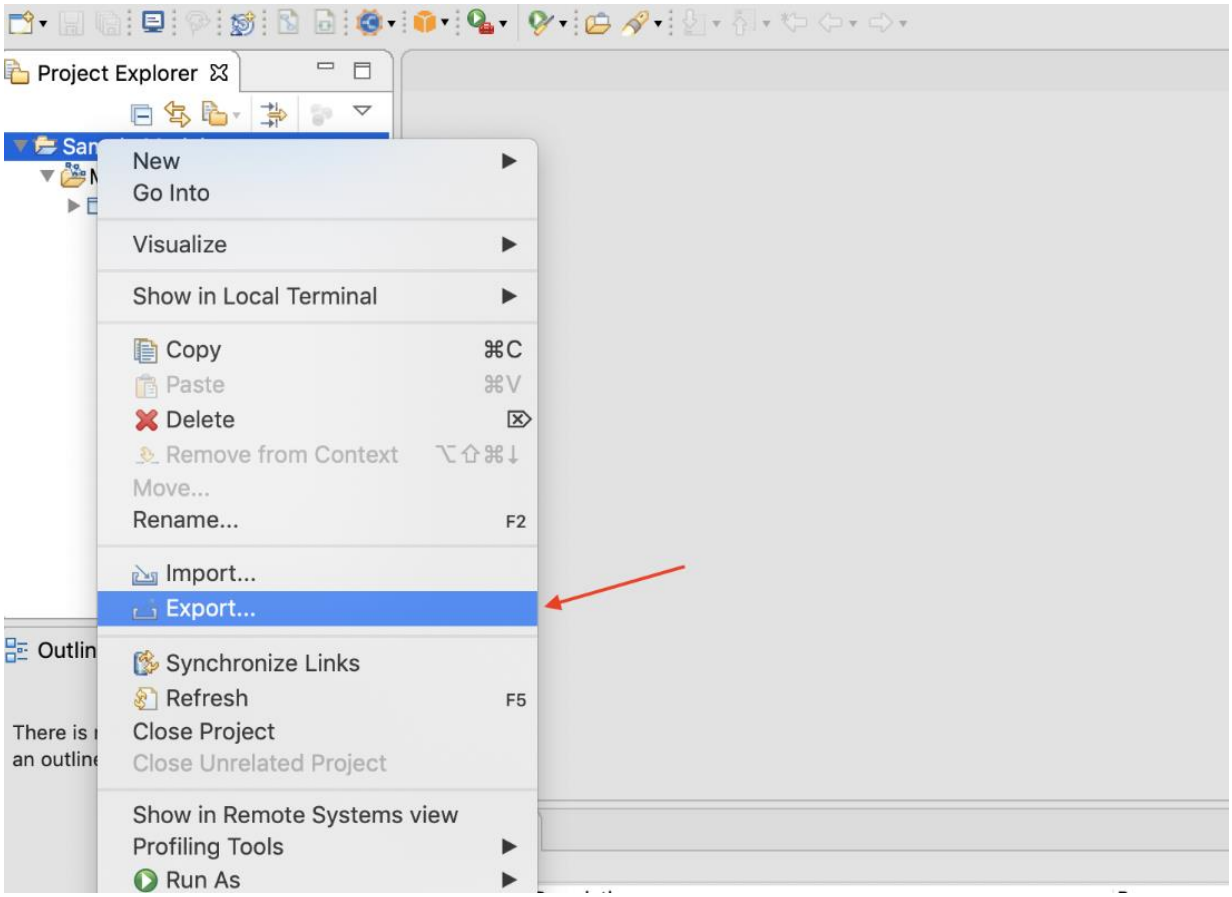
Exporting Models with Design Room ONE Integration Plugin

Below, we will highlight the steps to login in the eclipse client once Keycloak is enabled on the Design Room ONE server.

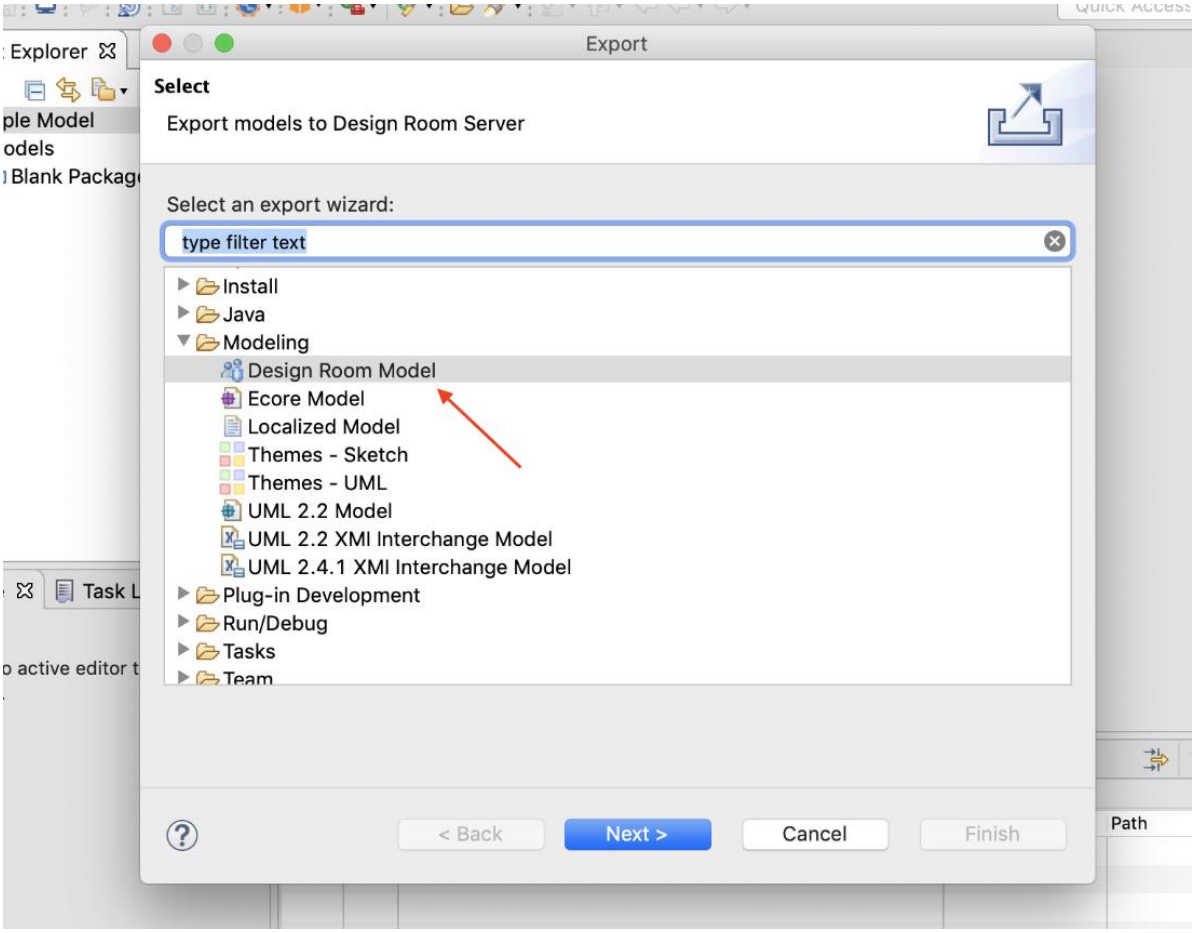
Prerequisites:

1. Install the Design Room ONE Integration feature in your modeling software by following the steps in the Design Room ONE installation document
2. Then ensure that the Design Room ONE server is started successfully.

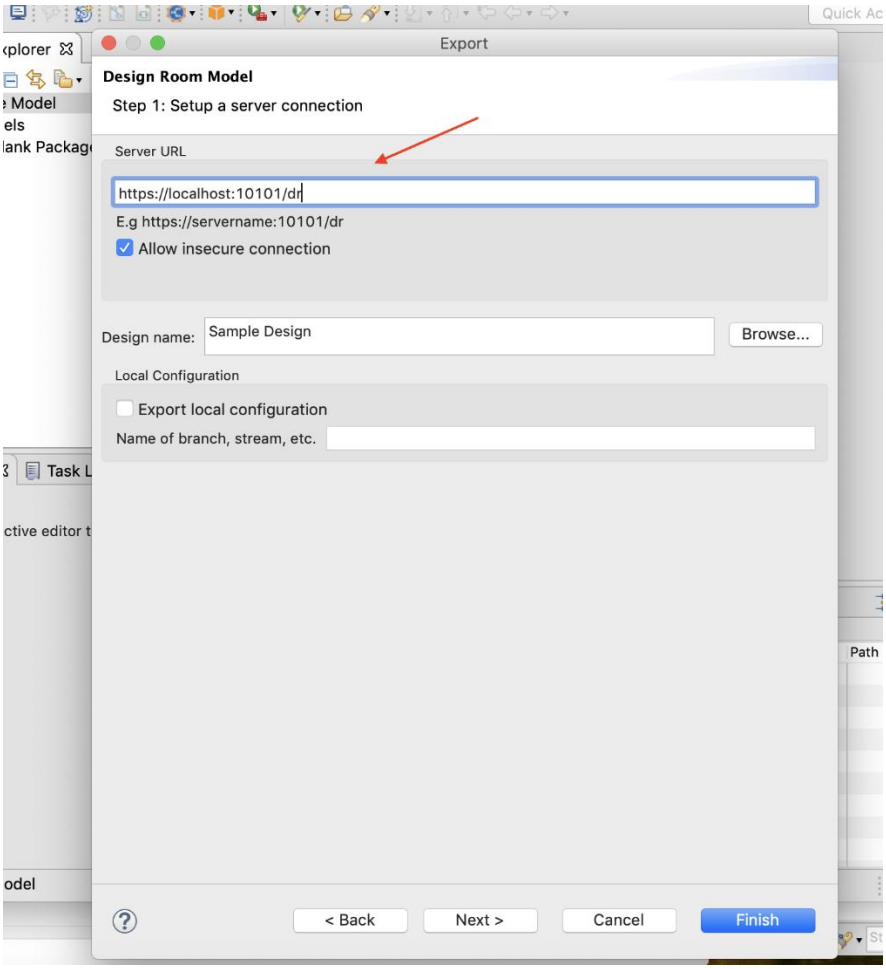
Right click on a project you want to export and select **Export**



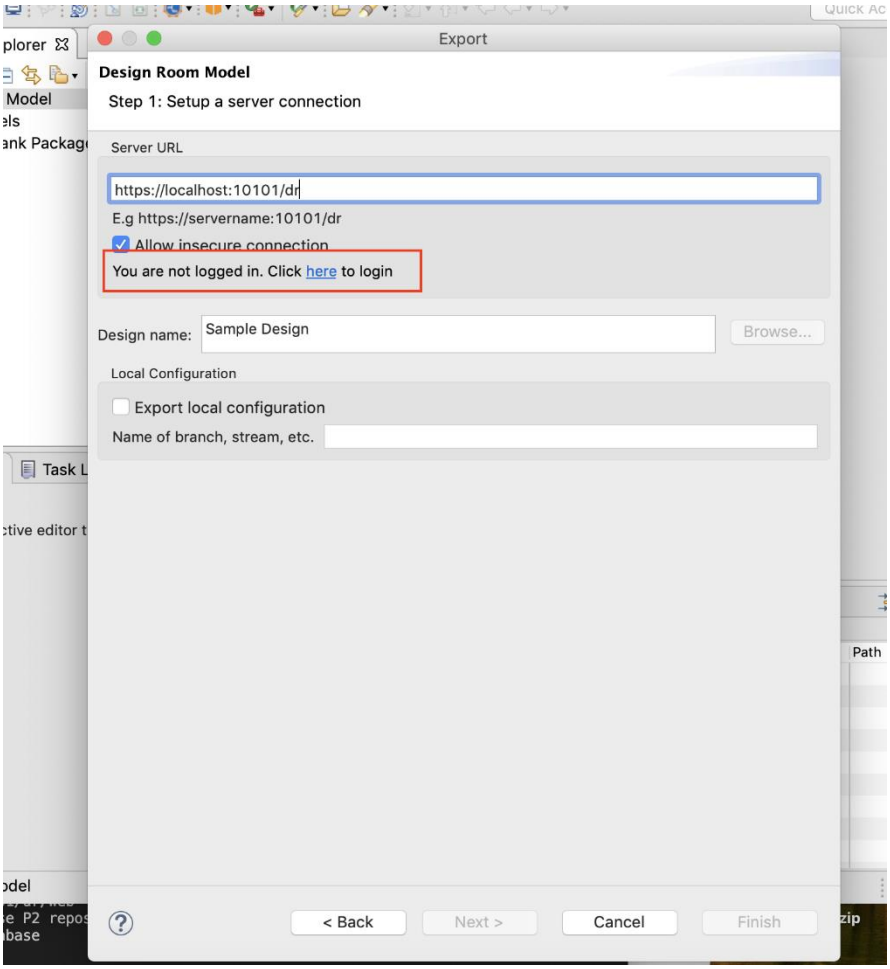
Under the **Modeling** folder, select **Design Room Model**



Then enter the server URL for your Design Room ONE server.



If authentication is enabled on the server and the URL is valid, you will see the message with a link to login.

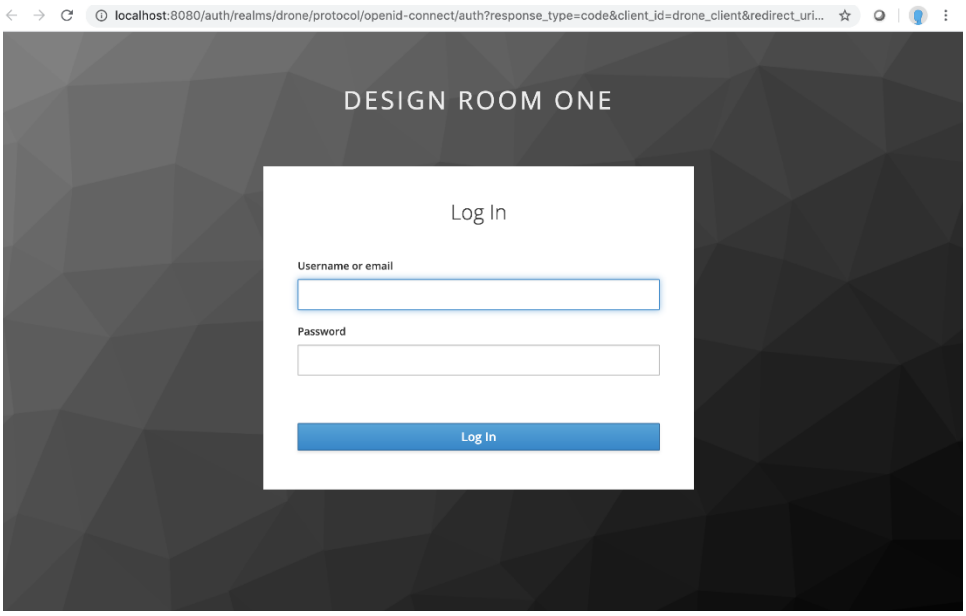


Note: If using RSAD with IBM Java sometimes users can get errors when connecting to Design Room ONE servers, for example if only TLS 1.2 is enabled on the server. To address this, it is recommended to make sure java version is at least **1.8_201** and add the following line in the end of **eclipse.ini** file in RSAD installation folder:

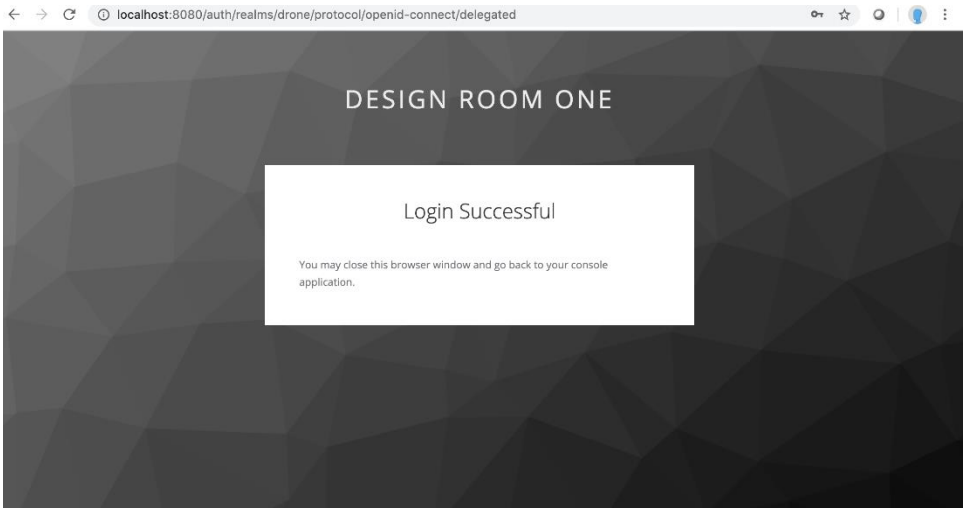
```
-Dcom.ibm.jsse2.overrideDefaultTLS=true
```

If Java version update cannot be performed an alternative solution could be adding the root certificate used to sign sandbox connection in the trust store of the JVM used by RSAD in addition to the change of **eclipse.ini** described above.

Once you click on hyperlink **here**, you will be redirected to the native browser to login

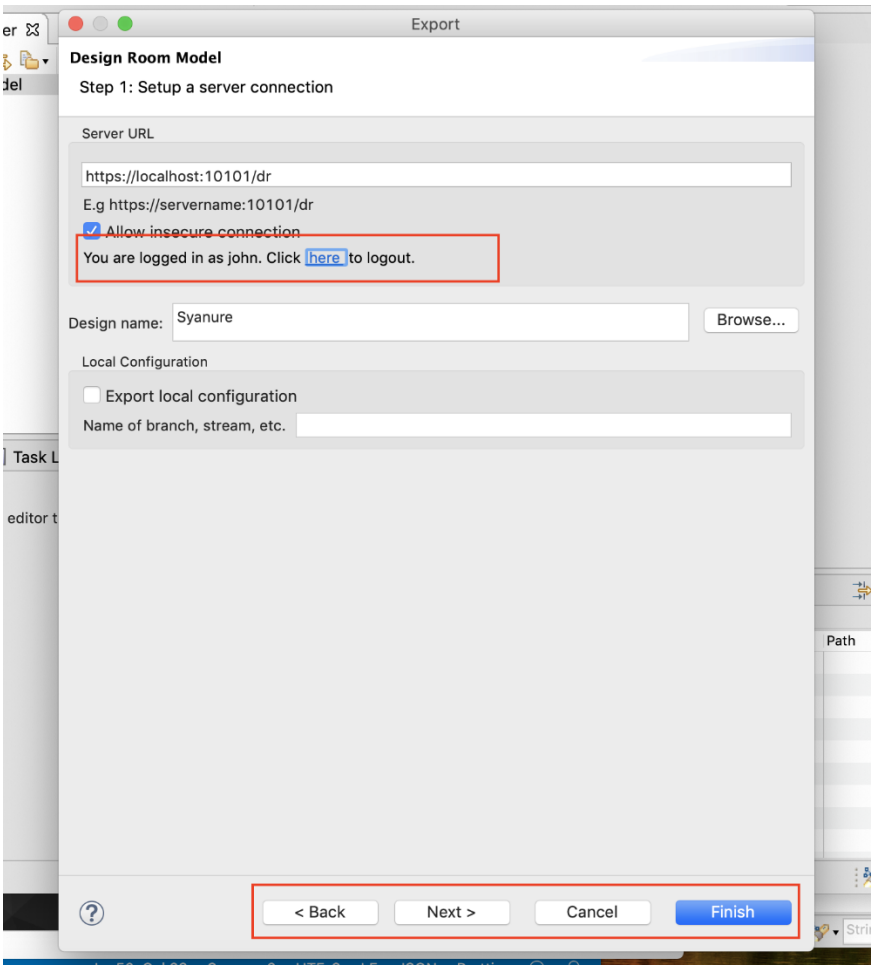


After a successful login, you will see the message.

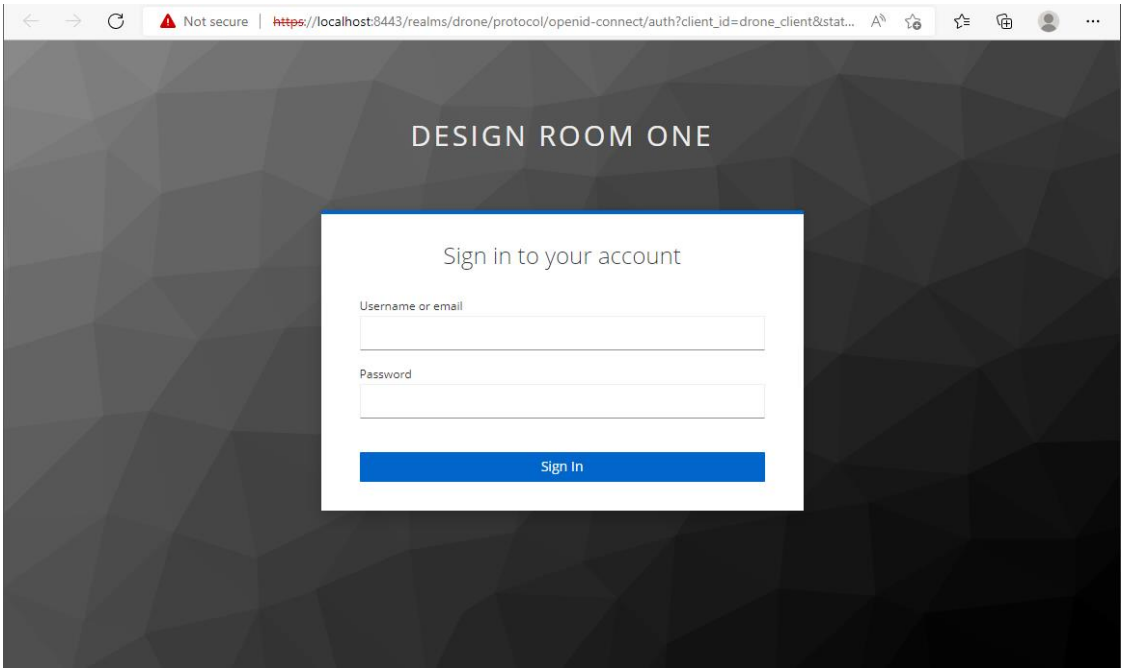


Note: Refer Known Limitations for Quarkus distribution.

Once you go back to your modeling tool, you will see that you are logged in and can proceed to make your export



To logout, you can click on the **here** link again and the window below will open in your native browser.



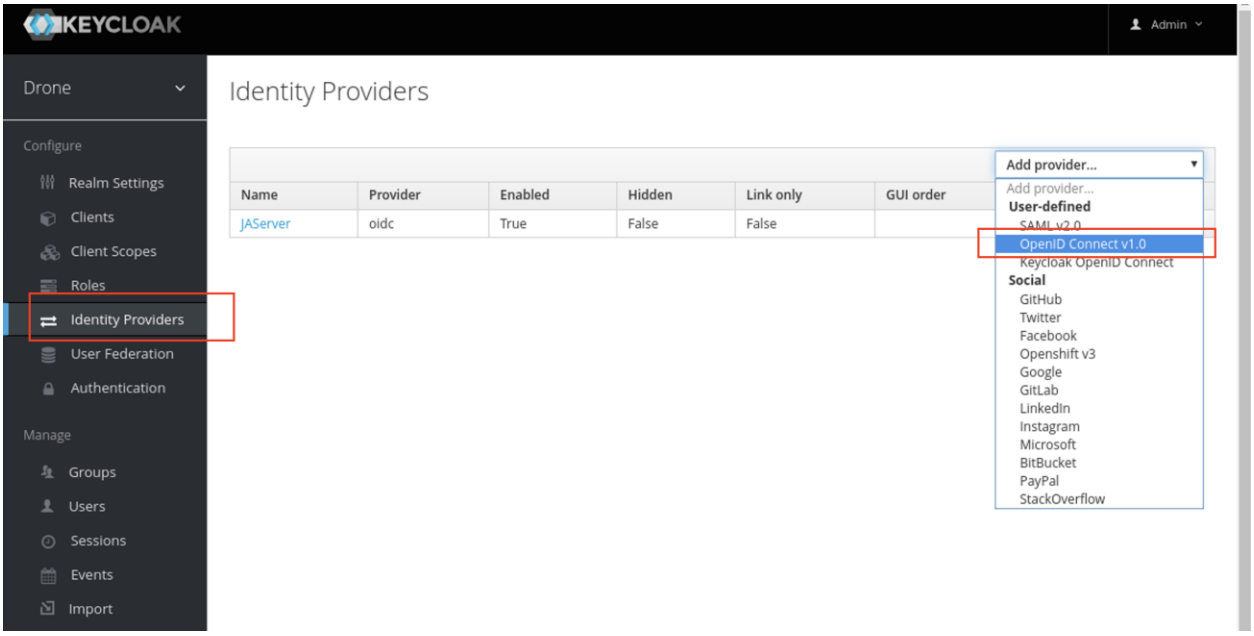
Single Sign-On with Jazz Authorization Server via OpenID

Prerequisites

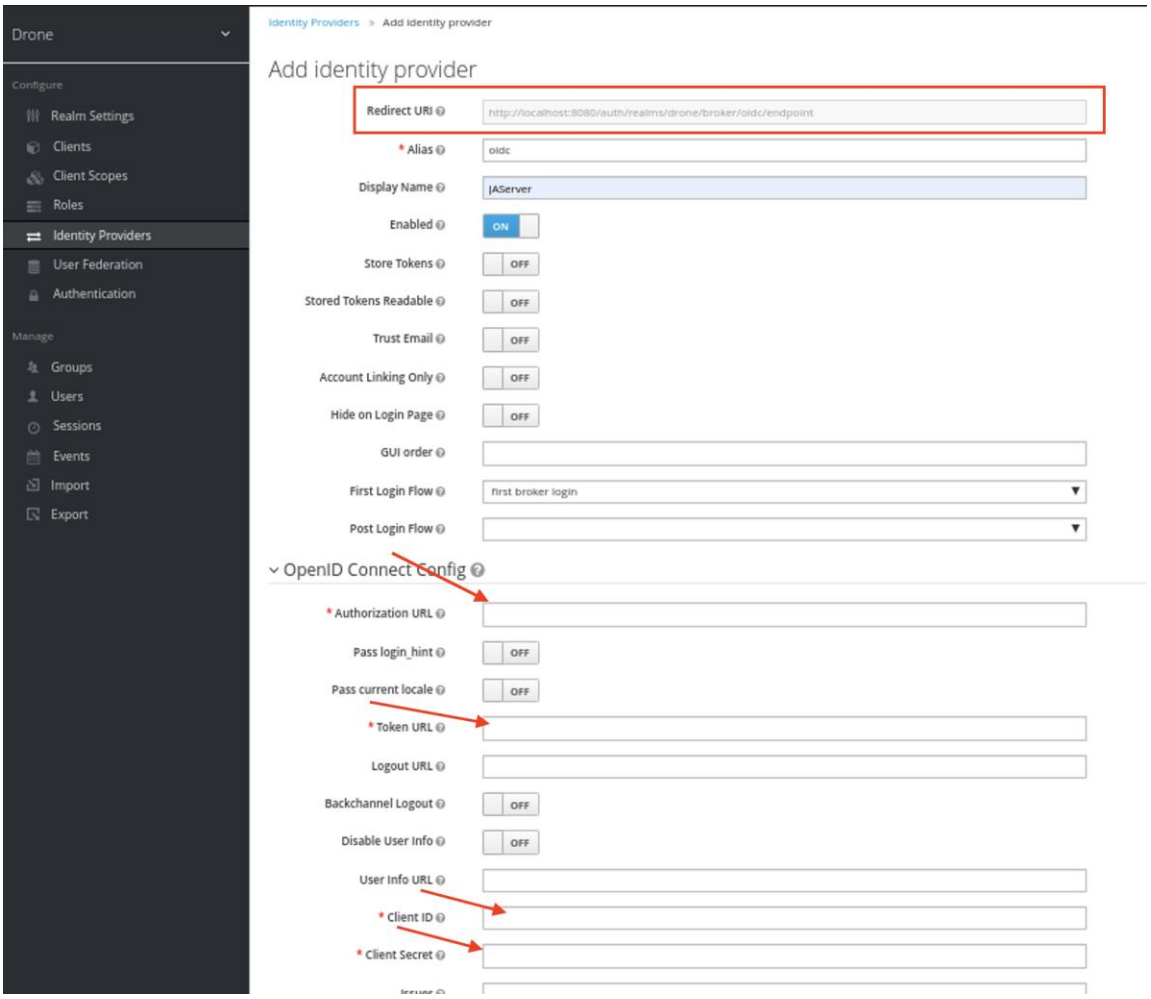
1. Jazz Authorization Server (JAS) is installed
2. Keycloak Server is installed

Creating a New Identify Provider in Keycloak

1. Login into Keycloak server as admin
2. Click on **DRONE** realm on the top left
3. Click on **Identity Providers** left menu open
4. Under the dropdown, select **OpenID Connect v1.0** as shown below



5. Create the alias name of your choice and use the redirect URI below to proceed in the next steps



6. You will notice that you are missing a few required fields such as “Authorization URL”, “Token URL”, “Client ID” and “Client Secret”. We will get these values from the jazz authorization server and complete the form later. Make note of the “Redirect URI” as we will need it in the next step.

Setting up JAS Configuration Security

If the JAS server is started with a self-signed certificate or no certificate at all, modify the **oauthProvider** element in the **appConfig.xml** file as shown below. Change the **httpsRequired** attribute to false.

```
<oauthProvider id="JazzOP"
    httpsRequired="false"
  autoAuthorize="true"
  customLoginURL="/jazzop/form/login"
  accessTokenLifetime="7201"
  authorizationGrantLifetime="604801">
  <autoAuthorizeClient>client01</autoAuthorizeClient>
    <databaseStore dataSourceRef="OAuthFvtDataSource" />
</oauthProvider>
```

Creating a Relying Party Application in JAS

1. Create a **body.json** file as shown below and make sure that you also add the above Keycloak redirect URI to the list of redirect_uris

```
{
  "token_endpoint_auth_method":"client_secret_basic",
  "scope":"openid profile email general",
  "grant_types":[
    "authorization_code",
    "client_credentials",
    "implicit",
    "refresh_token",
    "urn:ietf:params:oauth:grant-type:jwt-bearer"
  ],
  "response_types":[
    "code",
    "token",
    "id_token token"
  ],
  "application_type":"web",
  "subject_type":"public",
  "preauthorized_scope":"openid profile email general",
  "introspect_tokens":true,
  "trusted_uri_prefixes":[
    "https://keycloak.mycomp.any:*"
  ],
  "redirect_uris":[
    "http://keycloak.mycomp.any:8080/auth/realms/drone/broker/oidc/endpoint",
    "https://keycloak.mycomp.any:8443/auth/realms/drone/broker/oidc/endpoint"
  ]
}
```

Open command line and run the below command to create an application in the JAS server

```
curl --insecure --user admin:password --data @"./body.json"
http://jas.mycomp.any:9280/oidc/endpoint/jazzop/registration --header
"Content-Type: application/json"
```

2. If successful, the JAS server will respond with a response as shown below

```
{
   "client_id_issued_at":1583359157,
   "registration_client_uri":"https://localhost:9643/oidc/endpoint/jazzop/registration/d5852705ab4f4204ae29812373537277",
   "client_secret_expires_at":0,
   "token_endpoint_auth_method":"client_secret_basic",
   "scope":"openid profile email general",
   "grant_types":[
      "authorization_code",
      "client_credentials",
      "implicit",
      "refresh_token",
      "urn:ietf:params:oauth:grant-type:jwt-bearer"
   ],
   "response_types":[
      "code",
      "token",
      "id_token token"
   ],
   "application_type":"web",
   "subject_type":"public",
   "preauthorized_scope":"openid profile email general",
   "introspect_tokens":true,
   "trusted_uri_prefixes":[
      "https://localhost:*/"
   ],
   "resource_ids":[

   ],
   "client_id":"d5852705ab4f4204ae29812373537277",
   "client_secret":"p7Da1WpAxs0ujm8OmFn9pN2HDVXXtDEWXJy3pIa792TNgfUyGbU7tyKXPGSd",
   "client_name":"d5852705ab4f4204ae29812373537277",
   "redirect_uris":[
      "http://localhost:8080/auth/realms/drone/broker/oidc/endpoint",
      "https://localhost:8443/auth/realms/drone/broker/oidc/endpoint"
   ],
   "allow_regexp_redirects":false
}
```


3. Note that we now have a valid **client_id** and **client_secret** to complete the creation of our identity provider application in Keycloak. Please make a note of that **client_id** and **client_secret**.

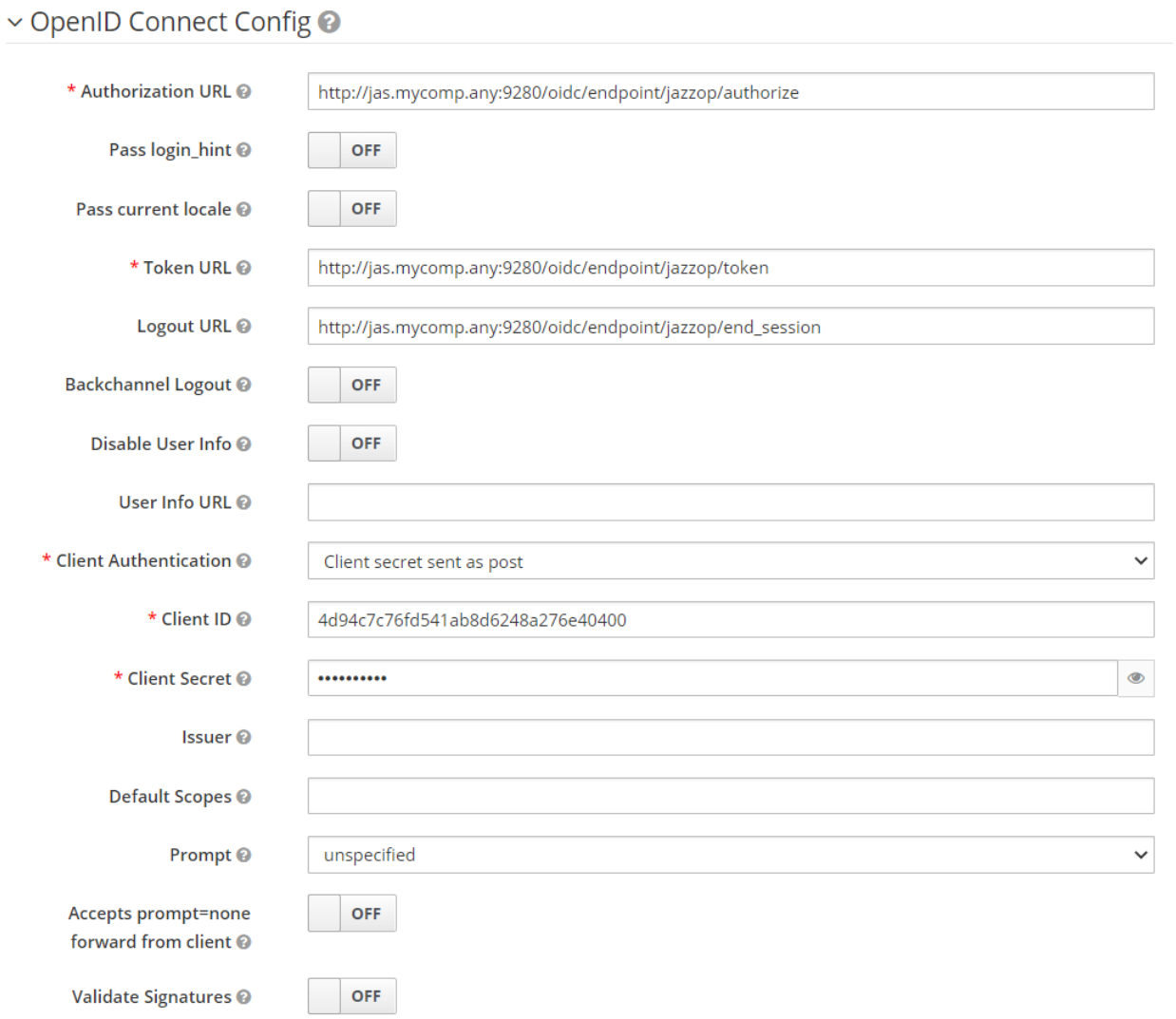
4. Browsing to the public endpoint http://jas.mycomp.any:9280/oidc/endpoint/jazzop/.well-known/openid-configuration will provide the remaining Authorization and token URLs of the JAS server assuming the JAS http server is started locally and on port 9280. A sample response is shown below.

```
{
   "introspection_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/introspect",
   "coverage_map_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/coverage_map",
   "issuer":"http://localhost:9280/oidc/endpoint/jazzop",
   "authorization_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/authorize",
   "token_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/token",
   "jwks_uri":"http://localhost:9280/oidc/endpoint/jazzop/jwk",
   "response_types_supported":[
      "code",
      "token",
      "id_token token"
   ],
   "subject_types_supported":[
      "public"
   ],
   "id_token_signing_alg_values_supported":[
      "HS256"
   ],
   "userinfo_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/userinfo",
   "registration_endpoint":"http://localhost:9280/oidc/endpoint/jazzop/registration",
   "scopes_supported":[
      "openid",
      "general",
      "profile",
      "email",
      "address",
      "phone"
   ],
   "claims_supported":[
      "sub",
      "groupIds",
      "name",
      "preferred_username",
      "picture",
      "locale",
```


5. We can now proceed back to Keycloak admin web page and finalize the identity provider form.

Creating a New Identify Provider in Keycloak Continued

6. Fill in the C**lient ID, Client Secret** and other fields as shown in the picture below.



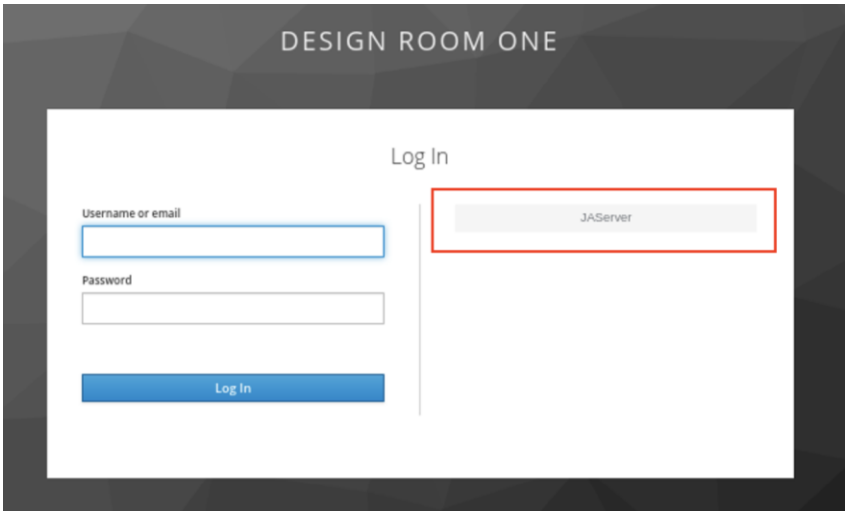
7. Make sure **Backchannel Logout** setting is off. After all required fields are completed, click the **Save** button.
8. Copy **Logout URL** and add it as the value "**clm_end_session"** property of **DR_ONE_INSTALL_DIR/OnPrem_Design_Room/config/server-config.json**
   The property should be placed inside **"dr_keycloak_config"** object

```
//Keycloak configuration
 "dr_keycloak_config": {
   "auth-server-url": "https://localhost:8443/auth",
   //Defines security level for Keycloak server
   //"none": HTTPS not required for any IP address
   //"external": Private IP and localhost can access without HTTPS
   //"all" : HTTPS required for all IP addresses
   "ssl-required": "external",
   "clm_end_session": https://jas.mycomp.any:9280/oidc/endpoint/jazzop/end_session
                        },
```

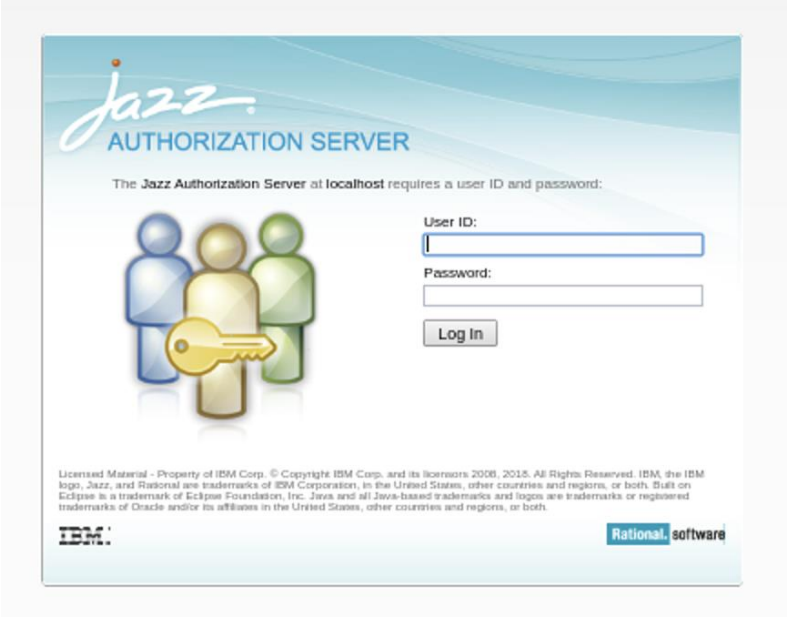
9. Restart Design Room ONE server to apply the configuration changes.

Logging in with JAS Authentication

1. Ensure that Keycloak is enabled in your Design Room ONE server (i.e., server-config.json) file. Start your Design Room ONE server and navigate to some URL e.g. https://drone.mycomp.any:10101/dr/web
   You should see the new identify provider option (JAServer) as shown here.



2. Click on JAServer option and you will be redirected to the JAS server to login



3. After a successful login you will be redirected back to the Keycloak server to add additional details for the user.

After a successful completion of the form, you will be directed back to the Design Room ONE page you navigated.

Known Limitations

1. In case of Quarkus distribution, after clicking on **login** link in the exporting wizard in a modeling tool a web browser with a message “We are sorry... Page not found” appears, the message should say “Login Successful”.