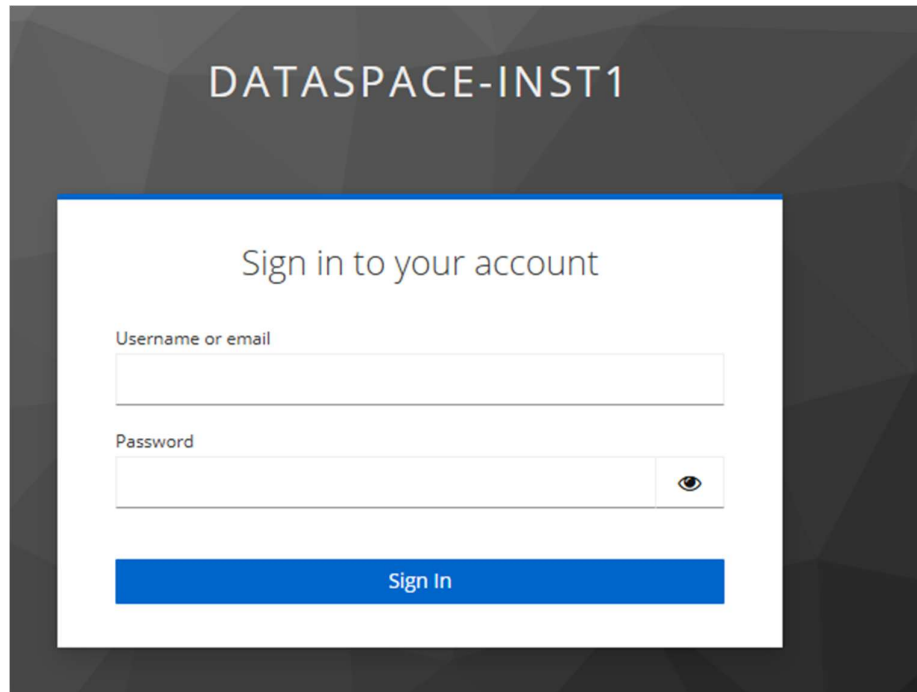
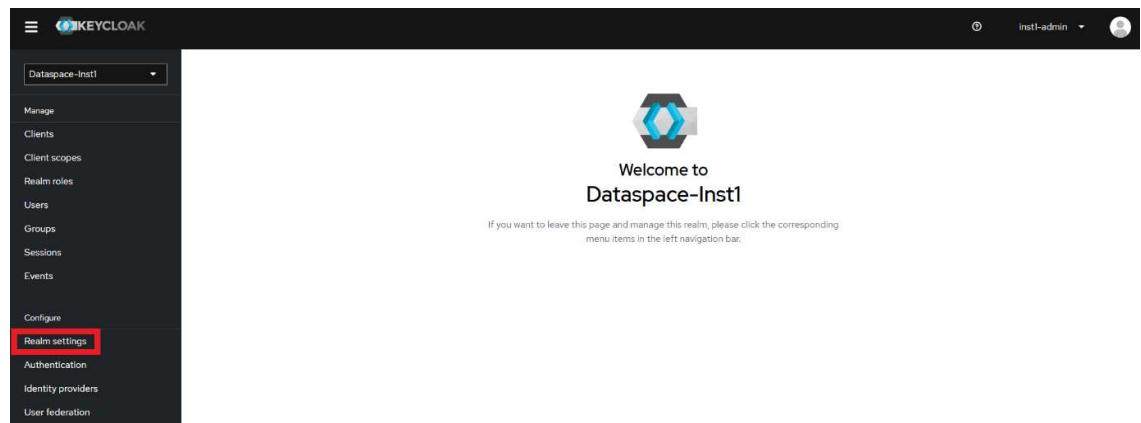


1. Sign into your Keycloak account

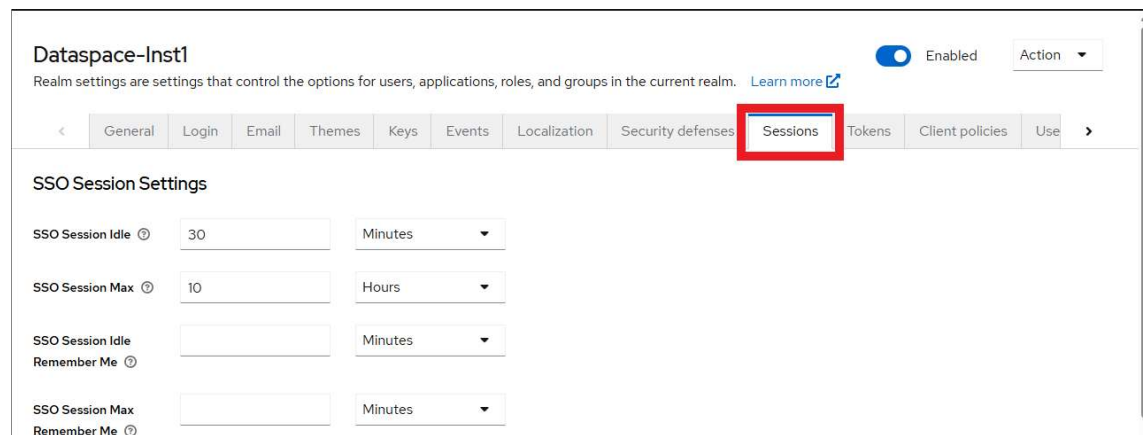


The image shows the Keycloak login interface for a realm named 'DATASPACE-INST1'. The page has a dark background with a white login box in the center. Inside the box, the text 'Sign in to your account' is displayed. Below this, there are two input fields: 'Username or email' and 'Password'. The password field includes a toggle icon for visibility. At the bottom of the box is a blue 'Sign In' button.

2. Click on **Realm settings** in the menu on the left-hand side



1. Click on **Sessions**



The image shows the 'Sessions' tab within the 'Dataspace-Inst1' realm settings. The 'Sessions' tab is highlighted with a red rectangle. The page displays 'SSO Session Settings' with several configuration options. Each option consists of a label, a text input field, and a unit dropdown menu. The 'SSO Session Idle' setting is currently set to 30 minutes. The 'SSO Session Max' setting is currently set to 10 hours. The 'SSO Session Idle Remember Me' and 'SSO Session Max Remember Me' settings are currently empty.

Setting	Value	Unit
SSO Session Idle	30	Minutes
SSO Session Max	10	Hours
SSO Session Idle Remember Me		Minutes
SSO Session Max Remember Me		Minutes

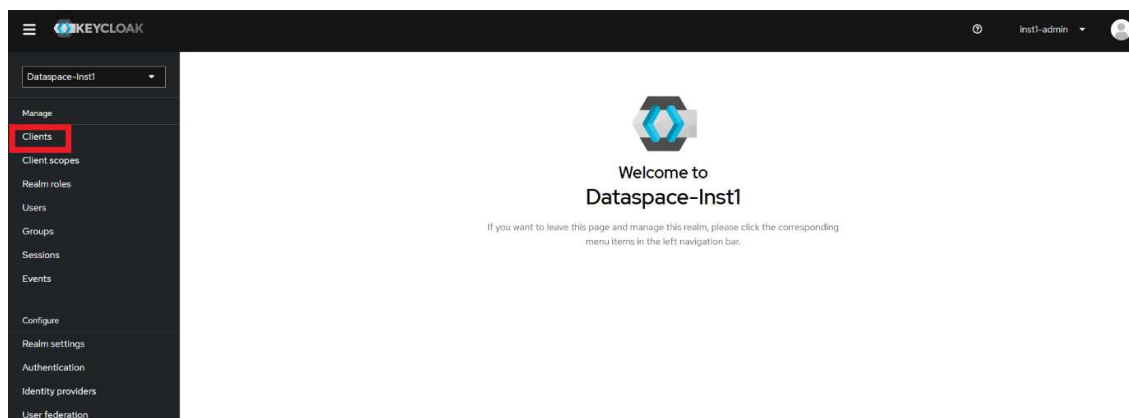
2. Set **Client Session Idle** to **30**

Client session settings

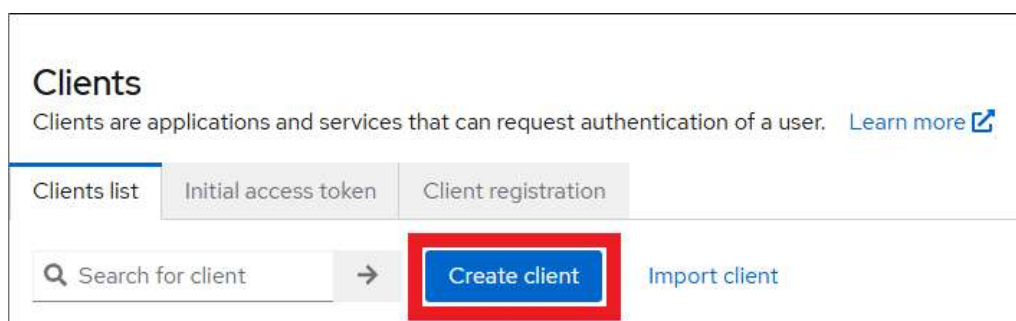
Client Session Idle ?	30	Minutes
Client Session Max ?		Minutes

3. Click on **Save**

3. Click on **Clients** in the menu on the left-hand side



1. Click on **Create client**



2. **Client ID: apikey-guard**

Create client

Clients are applications and services that can request authentication of a user.

1

General settings

Client type ⓘ

OpenID Connect

Client ID * ⓘ

apikey-guard

Name ⓘ

Description ⓘ

Always display in UI ⓘ

☐ Off

3. Click on **Next**

4. Turn **Client authentication** On

Create client

Clients are applications and services that can request authentication of a user.

2

Capability config

Client authentication ⓘ

☒ On

Authorization ⓘ

☐ Off

Authentication flow

☒ Standard flow ⓘ

☐ Implicit flow ⓘ

☐ OAuth 2.0 Device Authorization Grant ⓘ

☒ Direct access grants ⓘ

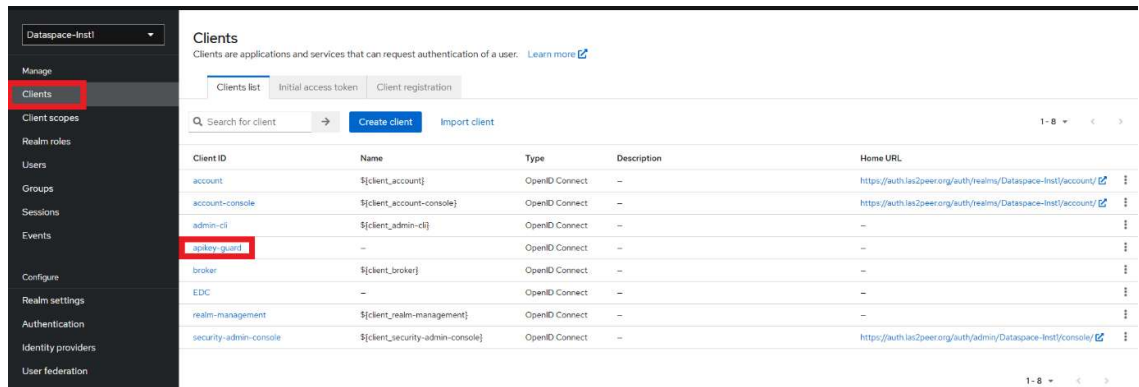
☐ Service accounts roles ⓘ

☐ OIDC CIBA Grant ⓘ

5. Click on **Next**

6. Click on **Save**

- Under **Clients**, click on the newly created entry **apikey-guard**



- In the Settings section, remove the entries for **Valid Redirect URIs** and **Web Origins**

Access settings

Root URL

Home URL

Valid redirect URIs

Add valid redirect URIs

Valid post logout redirect URIs

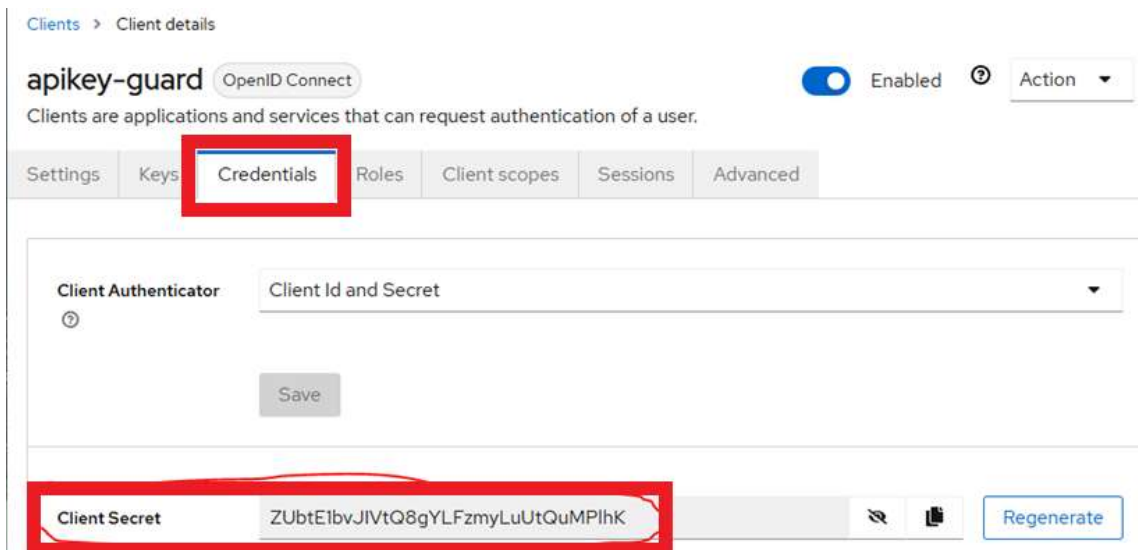
Add valid post logout redirect URIs

Web origins

Add web origins

i. Save

- In the **Credentials** section, copy the **Client Secret**



- Insert the **Client Secret** into the **compose_params.yaml** file of the DRK.

apikey_guard_verification_url:

"https://auth.las2peer.org/auth/realms/Dataspace-Inst1/protocol/openid-connect/token/introspect"

apikey_guard_client_id: "apikey-guard"

apikey_guard_client_secret:

"ZUbtE1bvJIVtQ8gYLFzmyLuUtQuMPlhK"

iam_url: <https://auth.las2peer.org/auth/realms/Dataspace-Inst1/protocol/openid-connect>

where **Dataspace-Inst1** is the Realm Prefix

```
apikey_guard_verification_url: "https://auth.las2peer.org/auth/realms/Dataspace-Inst1/protocol/openid-connect/token/introspect"
apikey_guard_client_id: "apikey-guard"
apikey_guard_client_secret: "ZUbtE1bvJIVtQ8gYLFzmyLuUtQuMPlhK"

iam_url: "https://auth.las2peer.org/auth/realms/Dataspace-Inst1/protocol/openid-connect"
```

5. In the **Clients** section, create a new entry.

Client ID	Name	Type	Description	Home URL
account	\$(client_account)	OpenID Connect	-	https://auth.las2peer.org/auth/realms/Dataspace-Inst1/account/
account-console	\$(client_account-console)	OpenID Connect	-	https://auth.las2peer.org/auth/realms/Dataspace-Inst1/account/
admin-cli	\$(client_admin-cli)	OpenID Connect	-	-
apikey-guard	-	OpenID Connect	-	-
broker	\$(client_broker)	OpenID Connect	-	-
EDC	-	OpenID Connect	-	-
realm-management	\$(client_realm-management)	OpenID Connect	-	-
security-admin-console	\$(client_security-admin-console)	OpenID Connect	-	https://auth.las2peer.org/auth/admin/Dataspace-Inst1/console/

1. **Client ID: EDC**

Create client

Clients are applications and services that can request authentication of a user.

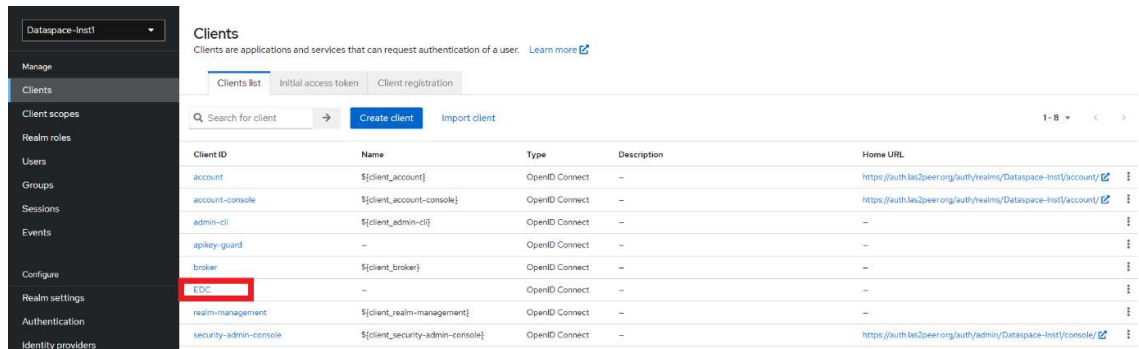
1 General settings

Client type ⓘ OpenID Connect

Client ID * ⓘ EDC

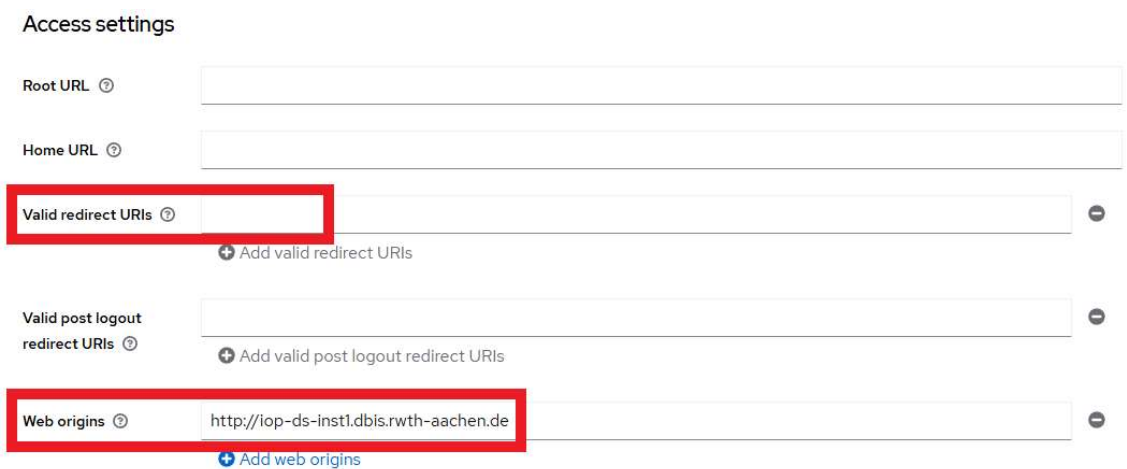
2. Click on **Next**, **Next**, **Save**

6. Under **Clients**, click on the newly created entry **EDC**



Client ID	Name	Type	Description	Home URL
account	`\${client_account}`	OpenID Connect	—	https://auth.las2peer.org/auth/realm/DataSpace-Inst1/account/
account-console	`\${client_account-console}`	OpenID Connect	—	https://auth.las2peer.org/auth/realm/DataSpace-Inst1/account/
admin-cli	`\${client_admin-cli}`	OpenID Connect	—	—
apikey-guard	—	OpenID Connect	—	—
broker	`\${client_broker}`	OpenID Connect	—	—
EDC	—	OpenID Connect	—	—
realm-management	`\${client_realm-management}`	OpenID Connect	—	—
security-admin-console	`\${client_security-admin-console}`	OpenID Connect	—	https://auth.las2peer.org/auth/admin/DataSpace-Inst1/console/

1. Delete the contents of **Valid Redirect URIs** and add the Domain of the Connector under **Web Origins**. In this case, the Domain is <https://iop-ds-inst1.dbis.rwth-aachen.de>



Access settings

Root URL

Home URL

Valid redirect URIs

Add valid redirect URIs

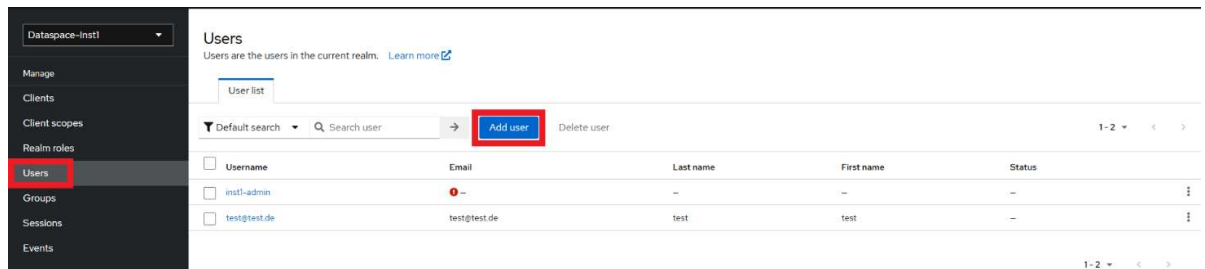
Valid post logout redirect URIs

Add valid post logout redirect URIs

Web origins <http://iop-ds-inst1.dbis.rwth-aachen.de>

Add web origins

2. Click on **Save**
7. In the **Users** section, create a new user account.



Username	Email	Last name	First name	Status
inst1-admin		—	—	—
test@test.de	test@test.de	test	test	—

1. For the **Username**, do not use spaces; it is better to use an email address. Enter the **Email** address and set **Email Verified** to **Yes**


Username *	<input type="text" value="test@test.de"/>
Email	<input type="text" value="test@test.de"/>
Email verified ⓘ	<input checked="" type="checkbox"/> Yes
First name	<input type="text" value="test"/>
Last name	<input type="text" value="test"/>

2. Click on **Create**

- i. In the **Credentials** section, set a password.

test@test.de Enabled Act

Details	Attributes	Credentials	Role mapping	Groups	Consents	Identity provider links	Sessions
---------	------------	-------------	--------------	--------	----------	-------------------------	----------



No credentials

This user does not have any credentials. You can set password for this user.

[Set password](#)

Set password for test@test.de

Password *



Password confirmation *



Temporary ⓘ

☐ Off

Save

Cancel

8. Rebuild the docker compose file on the Connector