Administration Service

Summary

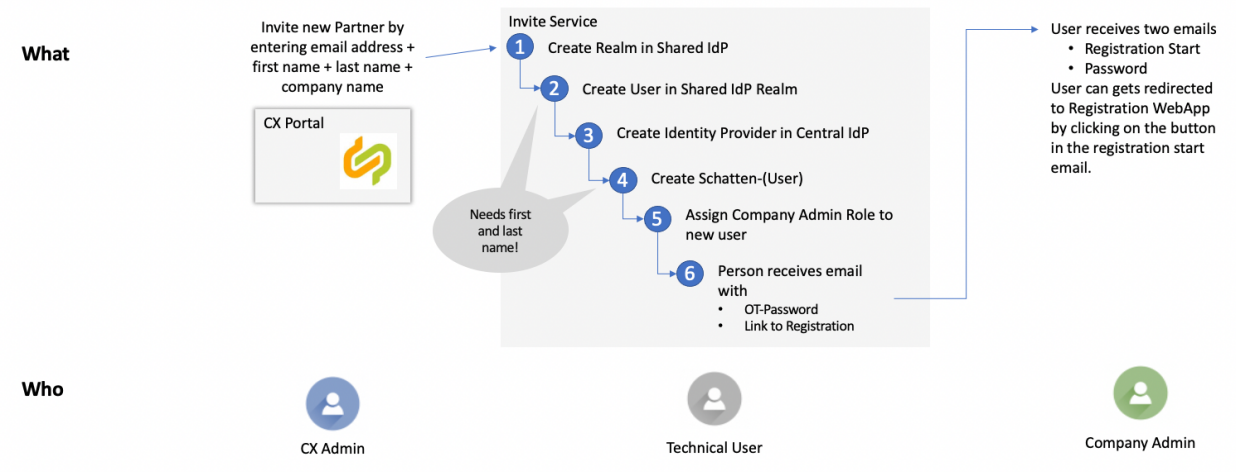
Administration service is the micro service for the company administration (e.g. users, setup, technical integration)

Invitation

The invitation is the starting point of a company onboarding.

A user of the company is getting invited by the CX Admin and the invitation endpoint is triggering in the backend the creation of the identity relevant elements.

The picture below shows this in a simple way



Invitation endpoint

FE is passing the necessary values - see invitation (POST) API in Swagger - which are getting used/processed by the backend.

POST-request must contain a valid bearer-token issued for central-realms client 'catenax-portal' with role 'invite_new_partner'.

body:

```
{
  "userName": "string",
  "firstName": "string",
```

```
  "lastName": "string",
  "email": "string",
  "organisationName": "string"
}
```

Endpoint: /invitation

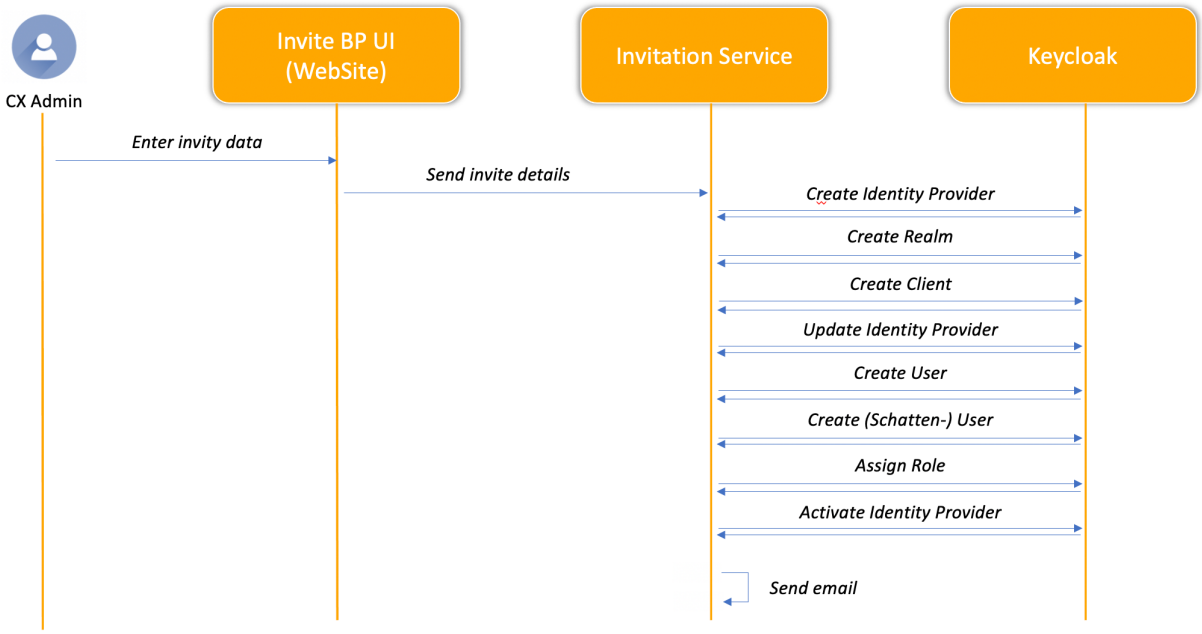
```
proposed endpoint fur versioning:
/api/administration/invitation/v2
```

With that information the service will start to create (with a technical user) the identity relevant elements in the CX IdP

- Realm
- User
- Identity Provider
- (Schatten-) User
- Role

As soon as those steps are executed, a email will get send to the invited user with the login details and the URL.

Architecture

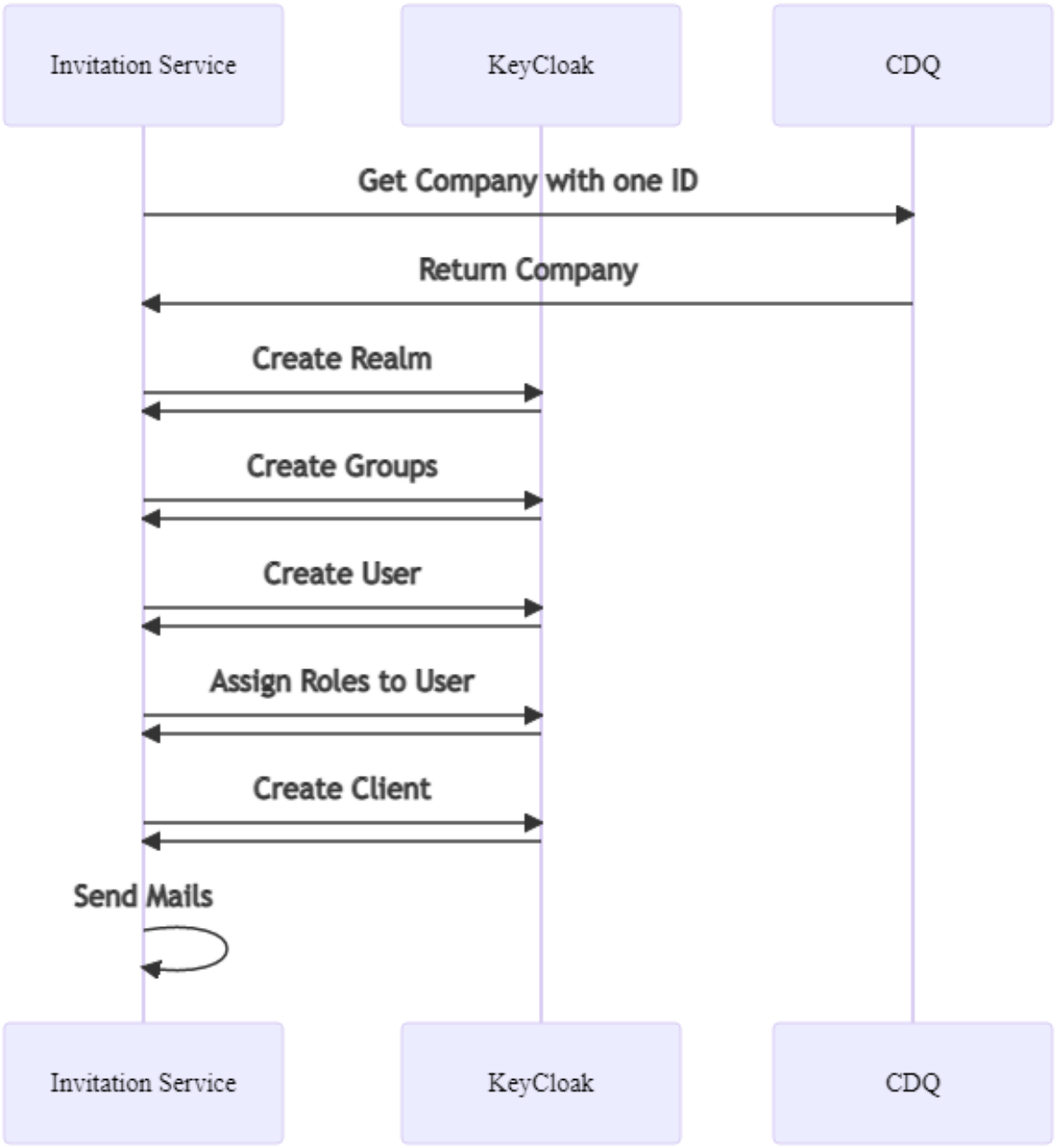


Catena-X_Flow diagram_invite service.pptx

implemented order of execution:

- validate user is eligible to execute service (request must contain valid token issued for client 'catenax-registration' with role 'CX Admin')
- determine next identity-providers name
- create identity-provider in disabled state to block that new idp-name
- create mapper-configurations for username, tenant (idp-name) and organisation in identityprovider config to synchronize those values with every user login
- retrieve oidc-metadata of central realm
- create new realm in shared keycloak
- create client for identityprovider in realm using metadata of central realm (redirect- and cert-url)
- retrieve oidc-metadata from new realm
- update identity-provider with metadata (auth-, token-, logout- and cert-url) from realm
- create user in shared realm
- create shadow-user in CX-Central realm
- add link to shared-realm user to the shadow-user
- assign role(s) to shadow-user
- activate identity-provider
- send emails

Click here to see the old Speedboat design



User

User creation endpoint

A user of the company is getting invited to the Catena-X Portal by a user who has the the role 'add_user_account' within the central-realms client 'catenax-portal', the endpoint is triggering in the backend the creation of the identity relevant elements.

FE is passing the necessary values - see users POST API in Swagger - which are getting used/processed by the backend.

Endpoint: /user/(tenant/{tenant}/users

Body (array to enable bulk user creation):

```
[
  {
    "userName": "string",
    "eMail": "string",
    "firstName": "string",
    "lastName": "string",
    "role": "string",
    "message": "string"
  }
]
```

Besides the body, the fronted has to pass the necessary "tenant" value (within the user token) which is getting used/processed by the backend.

With that information the endpoint will start to create (with a technical user) the relevant elements:

- User in the SharedIdP
- Identity Provider in CentralIdP
- (Schatten-) User in CentralIdP
- Role assignment in CentralIdP
- BPN assignment as user attribute in CentralIdP

As soon as those steps are executed, a e-mail will get send to the invited user with the login details and the URL.

The endpoint will provide a list of successfully created users by returning their e-mail addresses.

Implemented error handling

- if a user does already exist, the process isn't exited but continues with the next user
- if the requested client role for assignment doesn't exist, the process isn't exited but continues with the next user

Error Handling to be implemented

- return object of successfully and unsuccessfully created users with reason in case of failure
- case of non-existing client role, currently the user gets created but the unsuccessful role assignment cause for the e-mail not getting sent out
- validation of e-mail address and other values
- cancel process in case that there are passed >50 users

Show all users of my company (SharedIdP) endpoint

See users GET API in in Swagger:

Endpoint: /user/tenant/{tenant}/users

Authorization Policies:

- Check Tenant - Tenant (IdP name) passed by the frontend must match the one in the tenant claim of the user token.
- 'view_user_management' within the CX-Central realm client 'catenax-portal'

User deletion endpoint

See users DELETE API in in Swagger:

Endpoint: /user/tenant/{tenant}/users

Authorization Policies:

- Check Tenant - Tenant (IdP name) passed by the frontend must match the one in the tenant claim of the user token.
- 'delete_user_account' within the CX-Central realm client 'catenax-portal'

Own user deletion endpoint

See ownUser DELETE API in in Swagger:

Endpoint: /user/tenant/{tenant}/ownUser

Authorization Policy:

- Check Tenant - Tenant (IdP name) passed by the frontend must match the one in the tenant claim of the user token.

The relevant userId to delete is retrieved from the 'sub' claim of the user token.

Show all roles of an app/client endpoint

See roles GET API in in Swagger:

Endpoint: /user/client/{clientId}/roles

Authorization Policies:

- 'view_client_roles' within the CX-Central realm client 'catenax-portal'

BPN assignment as user attribute at Registration Approval (CX Admin)

See roles PUT API in in Swagger:

Endpoint: /user/company/{companyId}/bpnAtRegistrationApproval

Authorization Policies:

- 'approve_new_partner' within the CX-Central realm client 'catenax-portal'

BPN assignment as user attribute (Business Admin)

See roles PUT API in in Swagger:

Endpoint: /user bpn

Authorization Policies:

- 'modify_user_account' within the CX-Central realm client 'catenax-portal'

Registration

Company details (with address) endpoint

See users GET API in in Swagger:

Endpoint: /registration/application/{applicationId}/companyDetailsWithAddress

Authorization Policies:

- 'view_submitted_applications' within the CX-Central realm client 'catenax-portal'

Error Handling between Keycloak and Portal

In the unlucky scenario of a data inconsistency between portal db and the central IdP, an error handler is implemented which is resulting into a 500 internal service error for scenarios such as:

- app role not found in keycloak, but available in portal db
- idp/realm not found in keycloak, but available in portal db
- etc.

Please note, this scenario should usually not happen but got implemented for the case of inconsistent data to ensure a quick fix of the bug.

Important links

GitHub direction: https://github.com/catenax-ng/product-portal-backend/tree/main/src/useradministration/CatenaX.NetworkServices.UserAdministration.Service