GridGain

User Authentication in Distributed Applications

Denis Mekhanikov
Team Lead @ GridGain Cloud Team

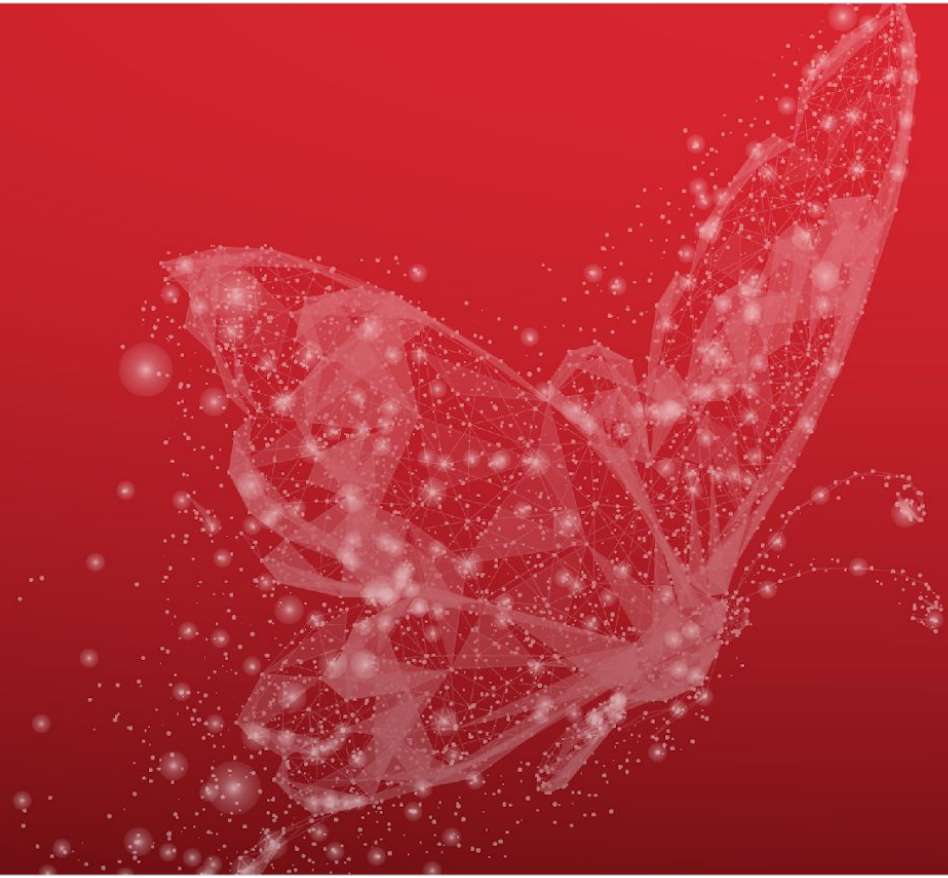
Alexander Kozhenkov
Software Engineer @ GridGain Cloud Team

Agenda

- Authentication models
- Single Sign On
- Example
  - GridGain and Control Center
- Demo
  - Integration of OpenID Connect with Control Center
  - SSO with secured cluster
  - Implementation of a custom Authenticator

GridGain

Authentication models

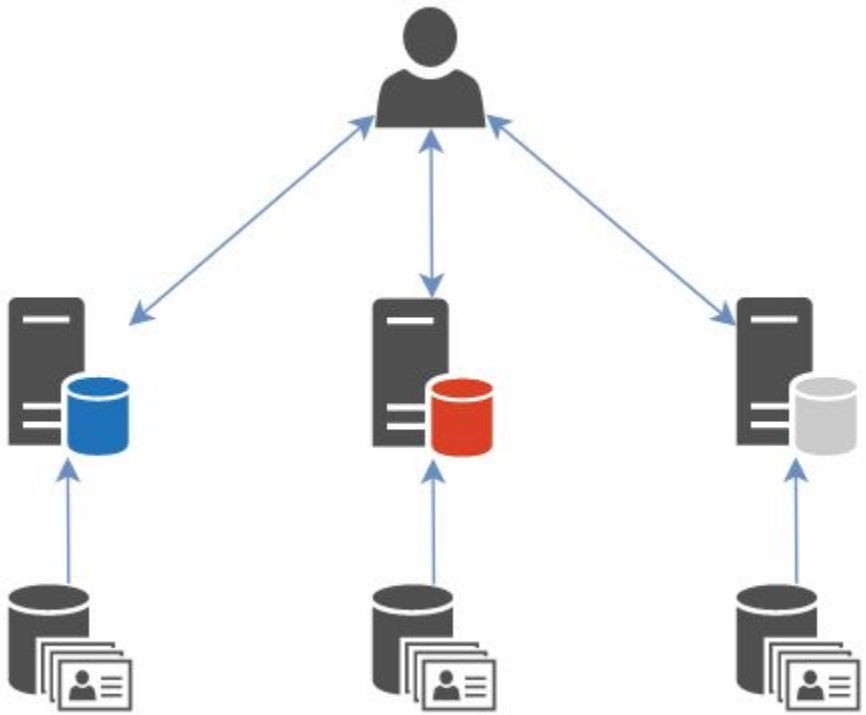


GridGain

Model #1:
Database of users inside application

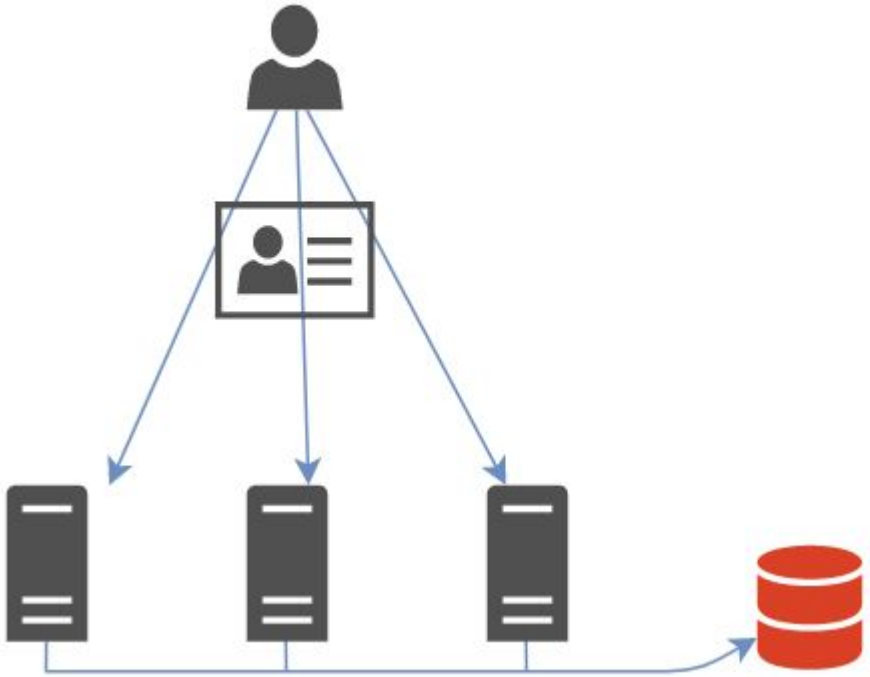
- No need for special interaction protocols
- An existing database can be used

Model #1:
Database of users inside application



- ! Syncronization issues between several applications
- ! Global user sessions are not supported

Model #2:
Shared database of users



- No need for synchronization
- Global sessions can be stored

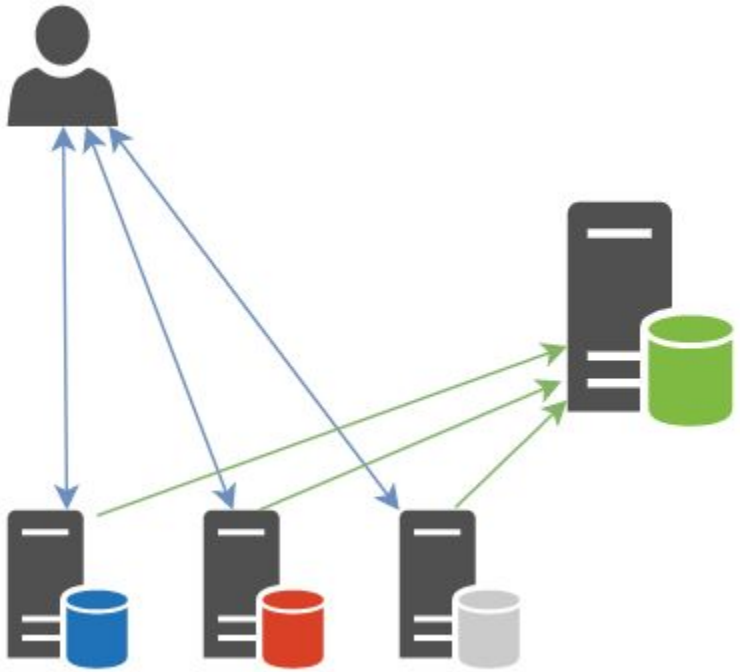
! Duplication of logic

! Problematic to make schema changes

! Only trusted applications can access the database

GridGain

Model #3:
Separate authentication service

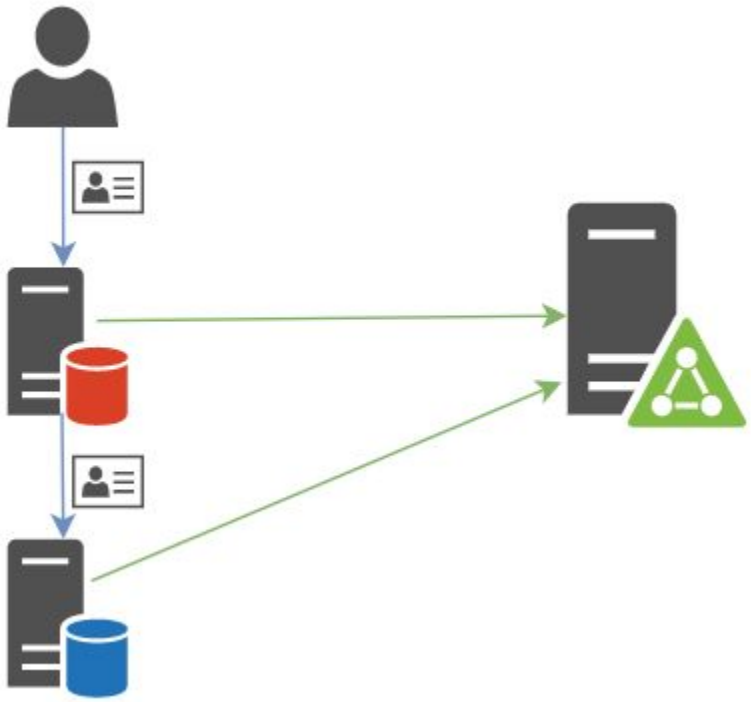


- Centralized database of users

! Need a special authentication protocol

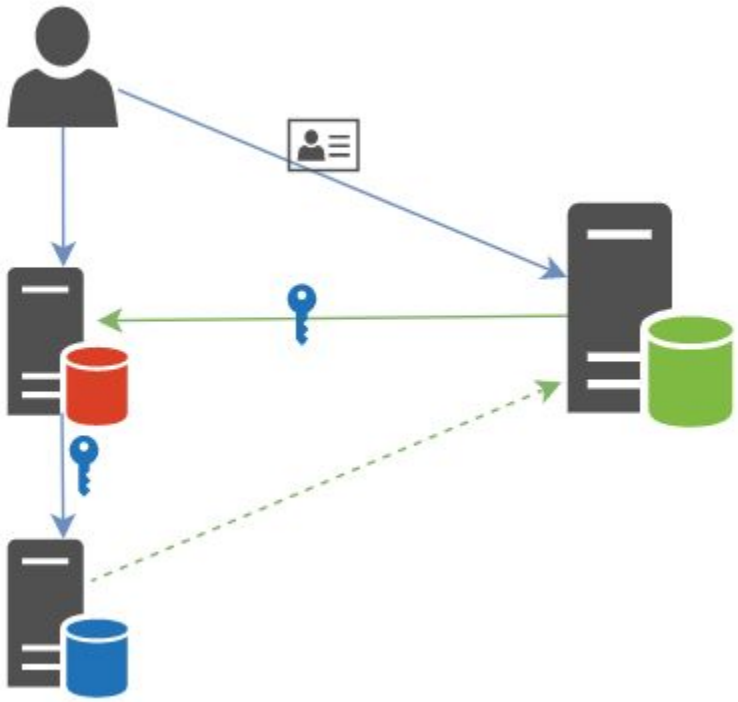
! Overhead on an additional external request

Model #3:
Separate authentication service



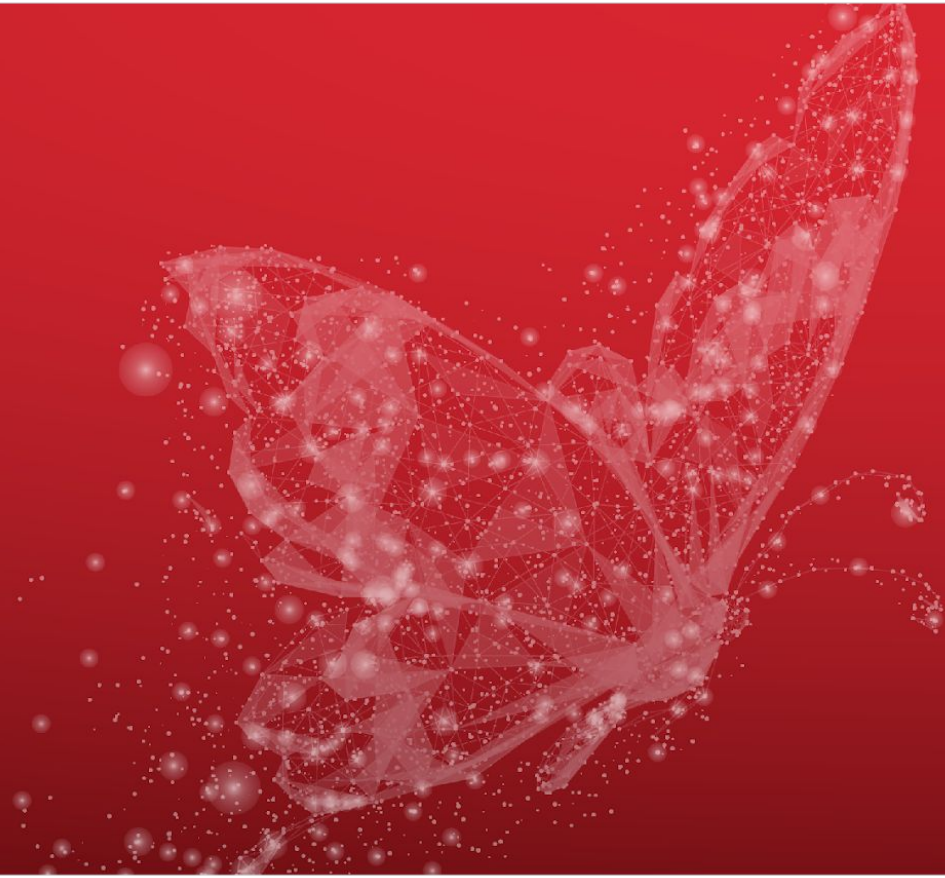
- Request to a new system – new authentication
- Propagation of users credentials between subsystems

Model #4: Single sign on



- Centralized database of users
- Single session for multiple applications
- Temporary token instead of a password

Single Sign On



GridGain

OAuth 2.0

OAuth 2.0 flows

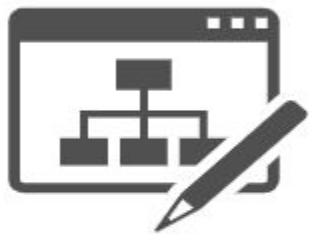
- Authorization code
- Implicit
- Client credentials
- Resource owner password
- Authorization code with proof key for code exchange

OAuth 2.0 flows

- Authorization code
- Implicit
- Client credentials
- Resource owner password
- Authorization code with proof key for code exchange

OAuth 2.0. Authorization code

PhotoEditor Frontend



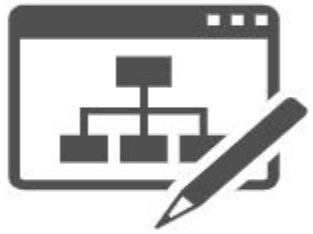
PhotoEditor Backend

Facebook

GridGain

OAuth 2.0. Authorization code

PhotoEditor Frontend

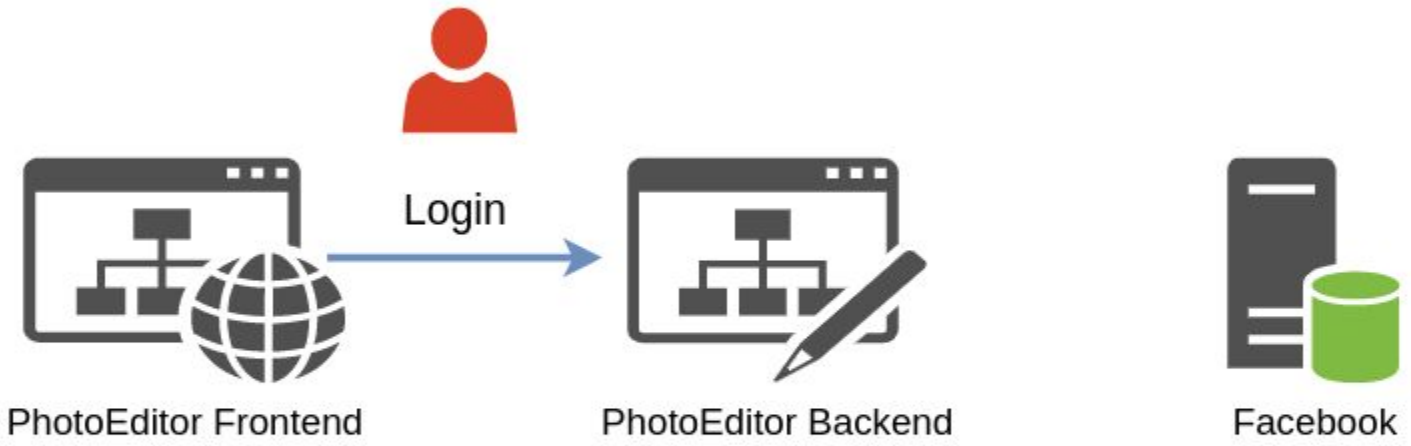


PhotoEditor Backend

Facebook

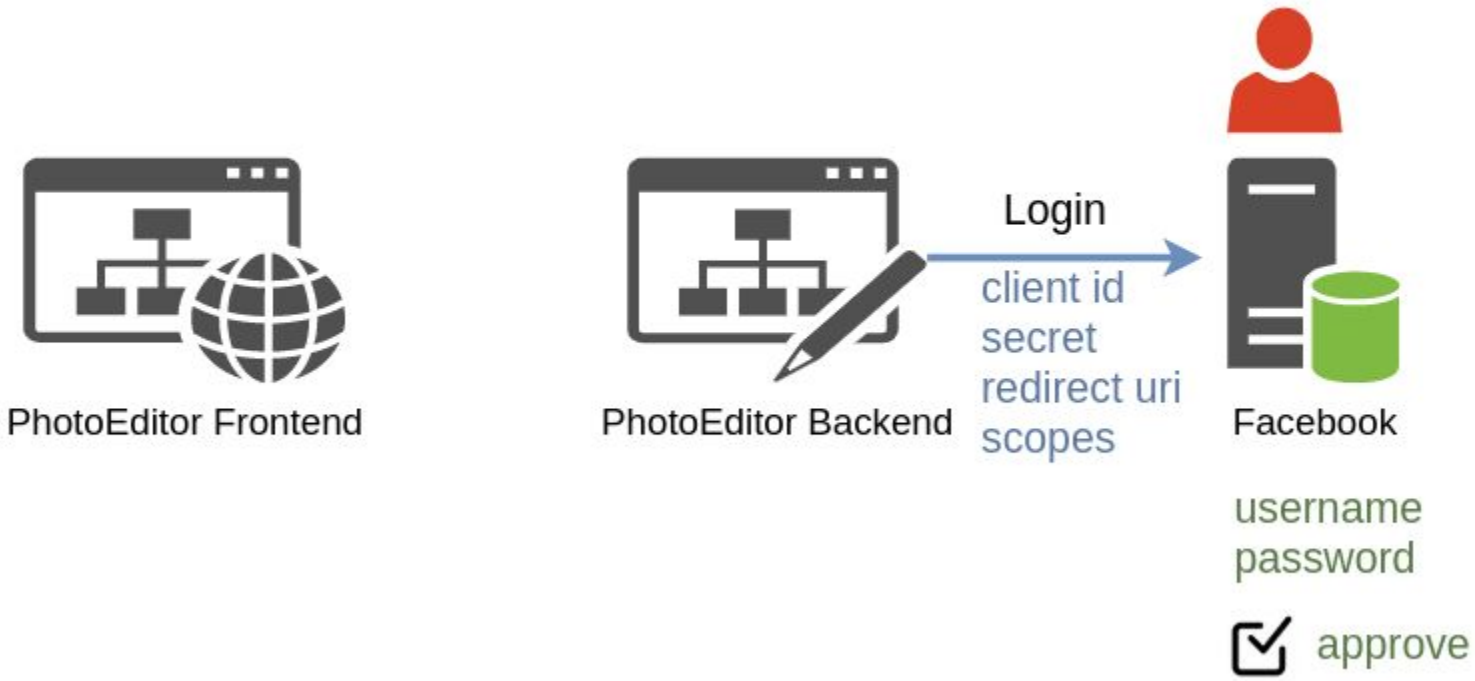
GridGain

OAuth 2.0. Authorization code



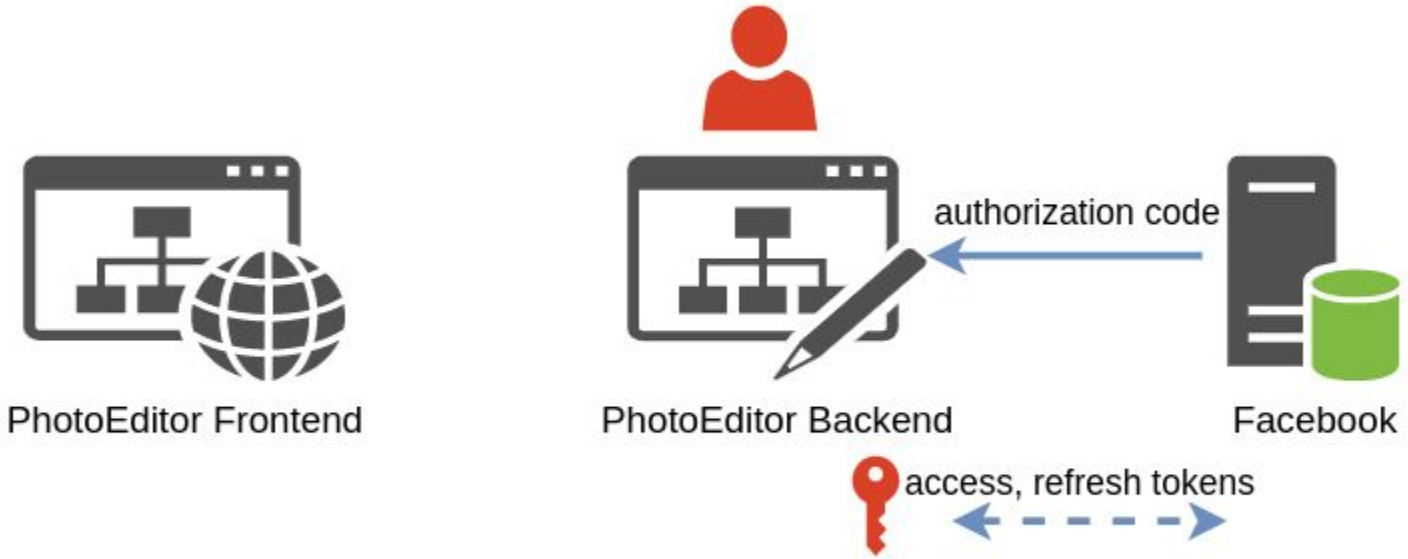
OAuth 2.0. Authorization code

OAuth 2.0. Authorization code

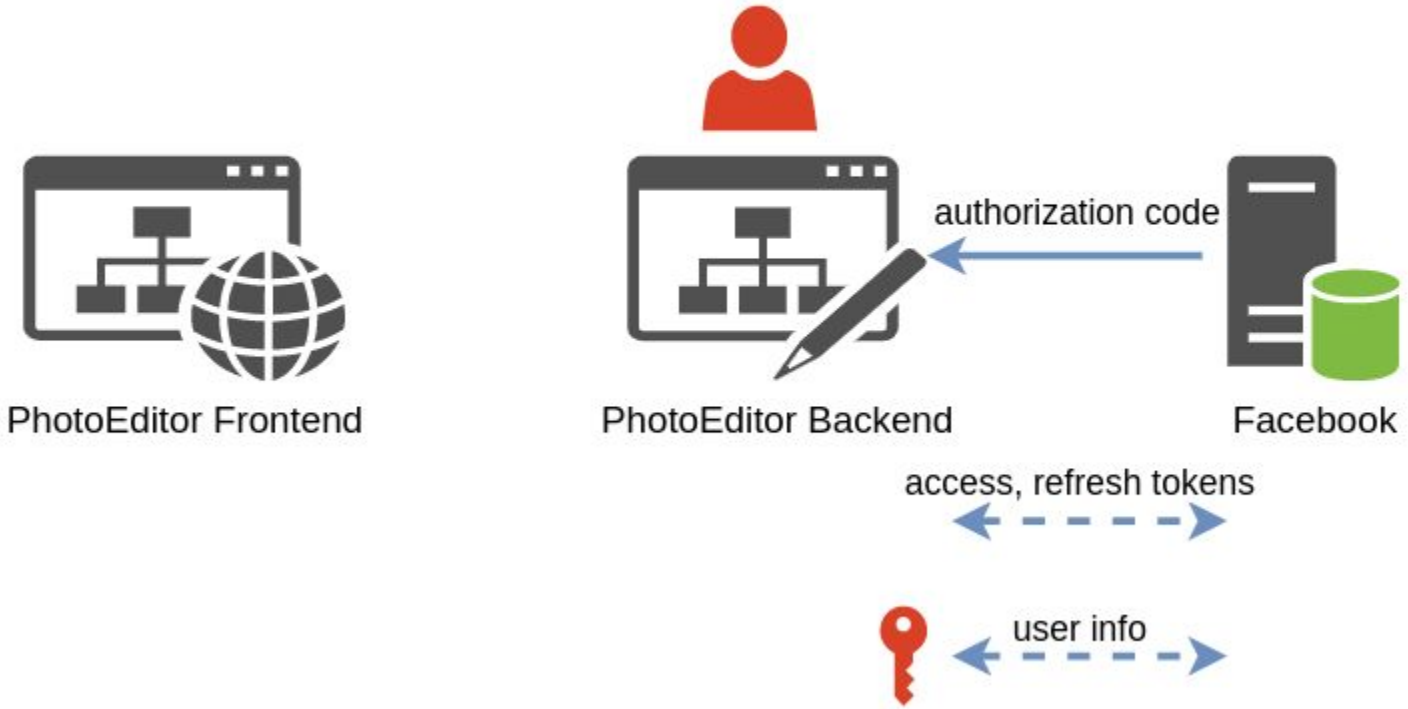


OAuth 2.0. Authorization code

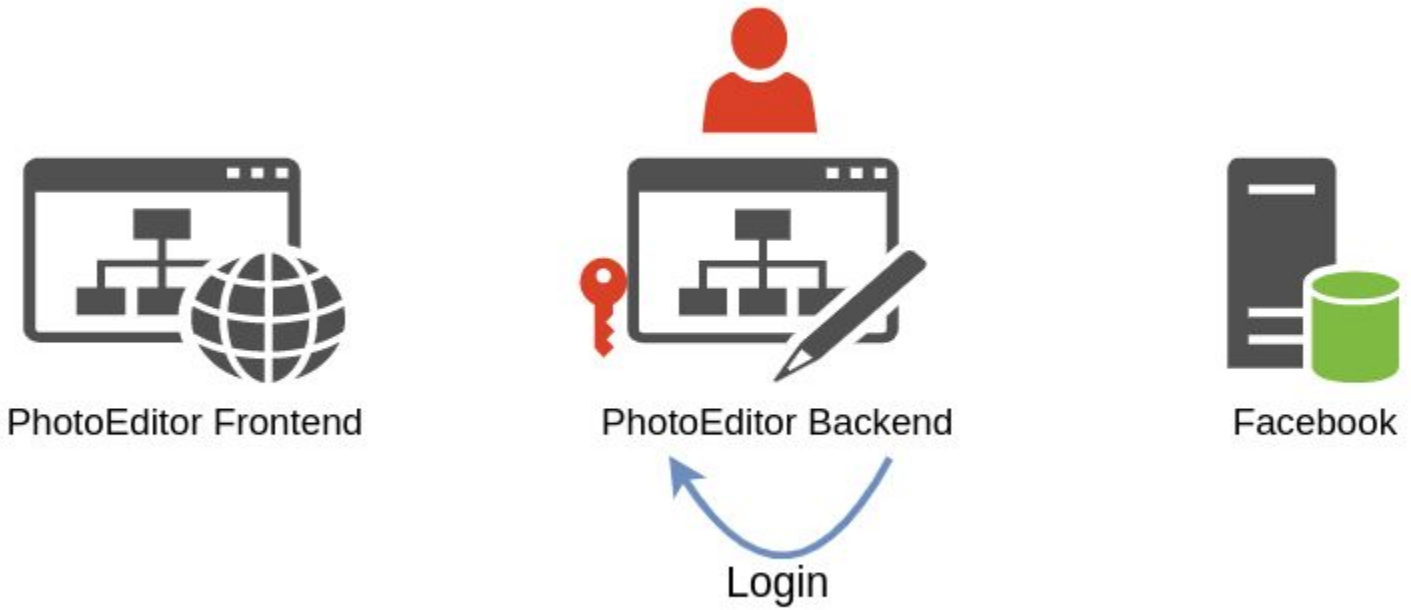
OAuth 2.0. Authorization code



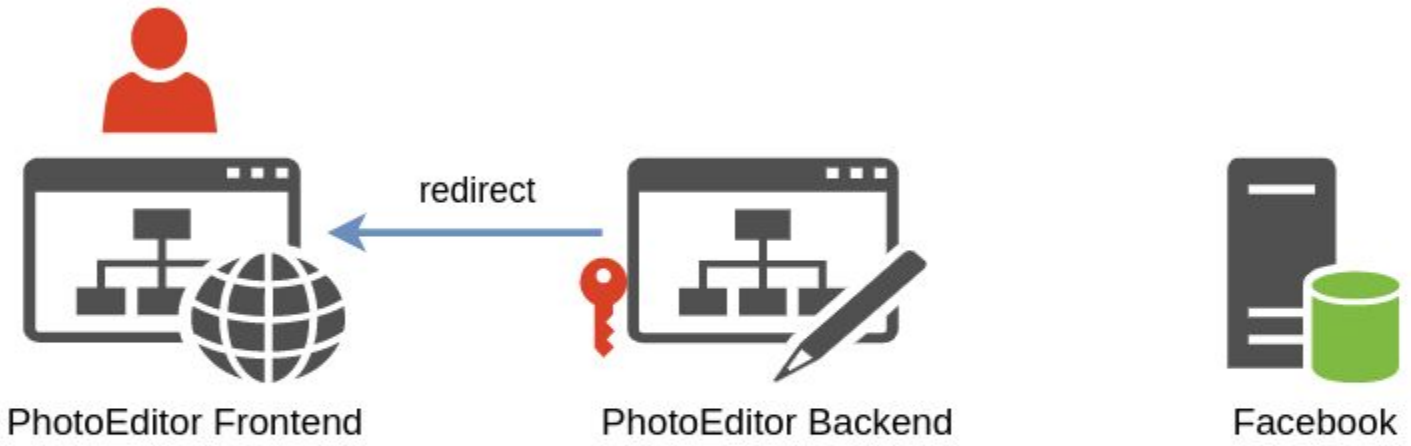
OAuth 2.0. Authorization code



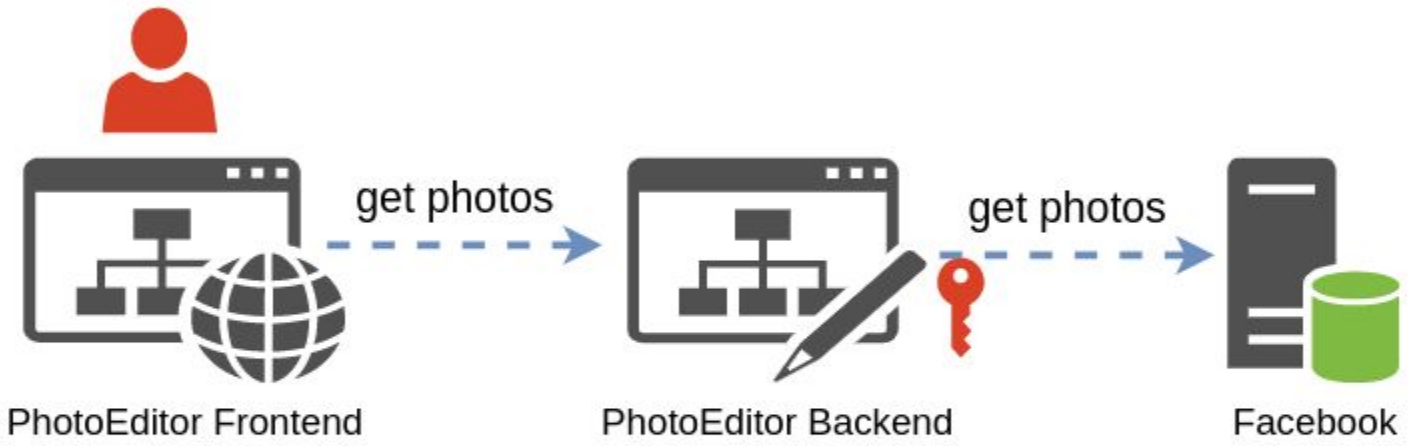
OAuth 2.0. Authorization code



OAuth 2.0. Authorization code



OAuth 2.0. Authorization code



OpenID Connect

OpenID Connect

- Extension of OAuth2.0

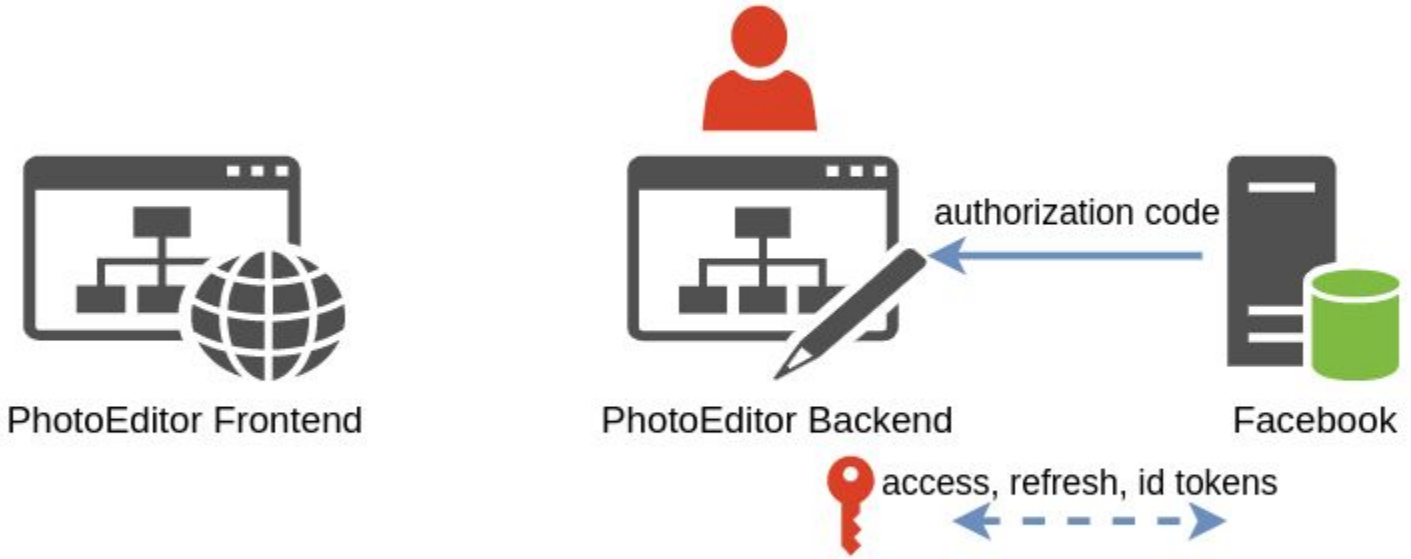
OpenID Connect

- Extension of OAuth2.0
- Authentication and authorization

OpenID Connect

- Extension of OAuth2.0
- Authentication and authorization
- The specification is more strict

OpenID Connect



OpenID Connect ID token

```
HTTP/1.1 200 OK
 Content-Type: application/json
 Cache-Control: no-store
 Pragma: no-cache

 {
  "access_token": "SlAV32hkKG",
  "token_type": "Bearer",
  "refresh_token": "8xLOxBtZp8",
  "expires_in": 3600,
  "id_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjFlOWdkazcifQ.ewogImlzc
    yI6ICJodHRwOi8vc2VydmVyLmV4YW1wbGUuY29tIiwKICJzdWIiOiAiMjQ4Mjg5
    NzYxMDAxIiwKICJhdWQiOiAiczZCaGRSa3F0MyIsCiAibm9uY2UiOiAibi0wUzZ
    fV3pBMk1qIiwKICJleHAiOiAxMzExMjgxOTcwLAogImlhdCI6IDEzMTEyODA5Nz
    AKfQ.ggW8hZ1EuVLuxNuuIJKX_V8a_OMXzR0EHR9R6jgdqrOOF4daGU96Sr_P6q
    Jp6IcmD3HP99Obi1PRs-cwh3LO-p146waJ8IhehcwL7F09JdijmBqkvPeB2T9CJ
    NqeGpe-gccMg4vfKjkM8FcGvnzZUN4_KSP0aAp1tOJ1zZwgjxqGByKHiOtX7Tpd
    QyHE5lcMiKPXfEIQILVq0pc_E2DzL7emopWoaoZTF_m0_N0YzFC6g6EJbOEoRoS
    K5hoDalrcvRYLSrQAZZKflyuVCyixEoV9GfNQC3_osjzw2PAithfubEEBLuVVk4
    XUVrWOLrLl0nx7RkKU8NXNHq-rvKMzqg"
 }
```


GridGain

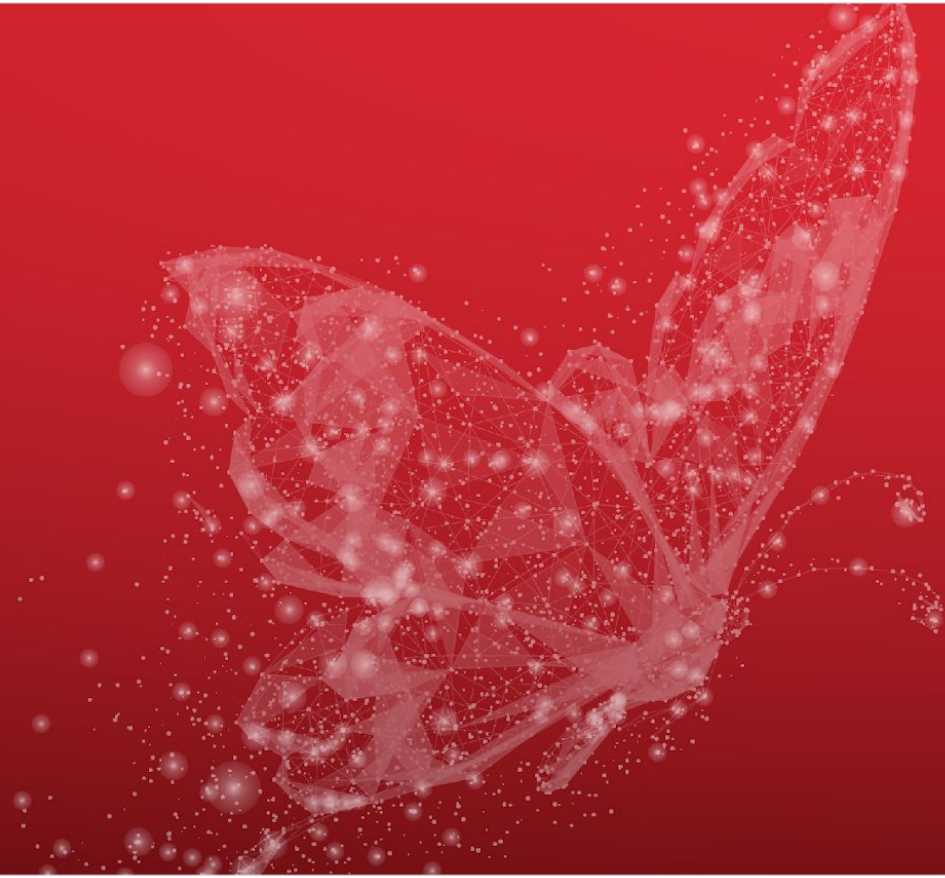
OpenID Connect ID token

```
HTTP/1.1 200 OK
 Content-Type: application/json
 Cache-Control: no-store
 Pragma: no-cache

 {
  "access_token": "SlAV32hkKG",
  "token_type": "Bearer",
  "refresh_token": "8xLOxBtZp8",
  "expires_in": 3600,
  "id_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjFlOWdkazcifQ.ewogImlzc
    yI6ICJodHRwOi8vc2VydmVyLmV4YW1wbGUuY29tIiwKICJzdWIiOiAiMjQ4Mjg5
    NzYxMDAxIiwKICJhdWQiOiAiczZCaGRSa3F0MyIsCiAibm9uY2UiOiAibi0wUzZ
    fV3pBMk1qIiwKICJleHAiOiAxMzExMjgxOTcwLAogImlhdCI6IDEzMTEyODA5Nz
    AKfQ.ggW8hZ1EuVLuxNuuIJKX_V8a_OMXzR0EHR9R6jgdqrOOF4daGU96Sr_P6q
    Jp6IcmD3HP99Obi1PRs-cwh3LO-p146waJ8IhehcwL7F09JdijmBqkvPeB2T9CJ
    NqeGpe-gccMg4vfKjkM8FcGvnzZUN4_KSP0aAp1tOJ1zZwgjxqGByKHiOtX7Tpd
    QyHE5lcMiKPXfEIQILVq0pc_E2DzL7emopWoaoZTF_m0_N0YzFC6g6EJbOEoRoS
    K5hoDalrcvRYLSrQAZZKflyuVCyixEoV9GfNQC3_osjzw2PAithfubEEBLuVVk4
    XUVrWOLrLl0nx7RkKU8NXNHq-rvKMzqg"
 }
```

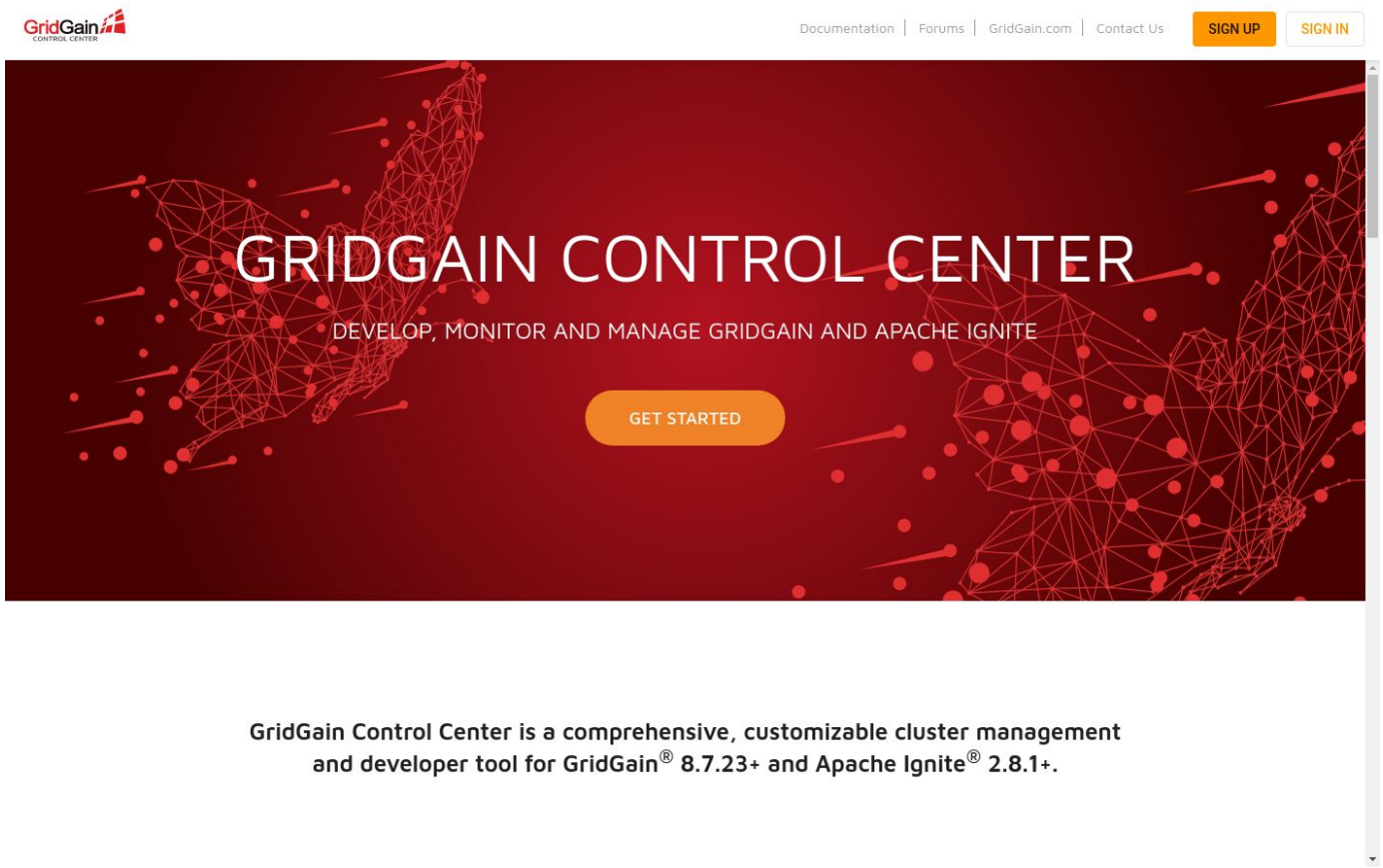
```
{
 "iss": "http://server.example.com",
 "sub": "248289761001",
 "aud": "s6BhdRkqt3",
 "nonce": "n-0S6_WzA2Mj",
 "exp": 1311281970,
 "iat": 1311280970,
 "name": "Jane Doe",
 "given_name": "Jane",
 "family_name": "Doe",
 "gender": "female",
 "birthdate": "0000-10-31",
 "email": "janedoe@example.com",
 "picture": "http://example.com/me.jpg"
}
```

Control Center



GridGain

Control Center overview



https://control.gridgain.com

Control Center overview

- Monitoring and management

https://control.gridgain.com

Control Center overview

- Monitoring and management
- Hosted, on-premise

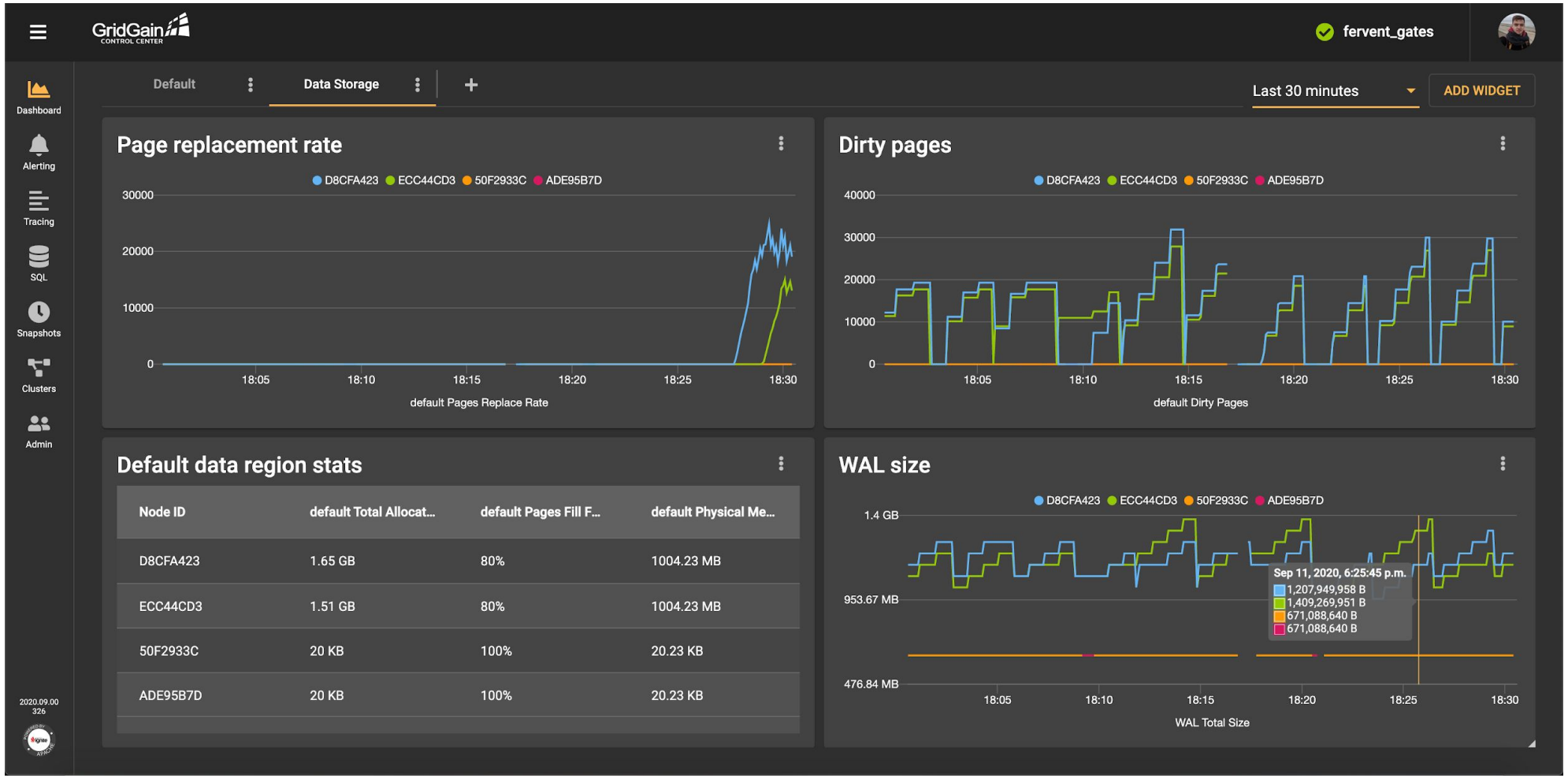
https://control.gridgain.com

Control Center overview

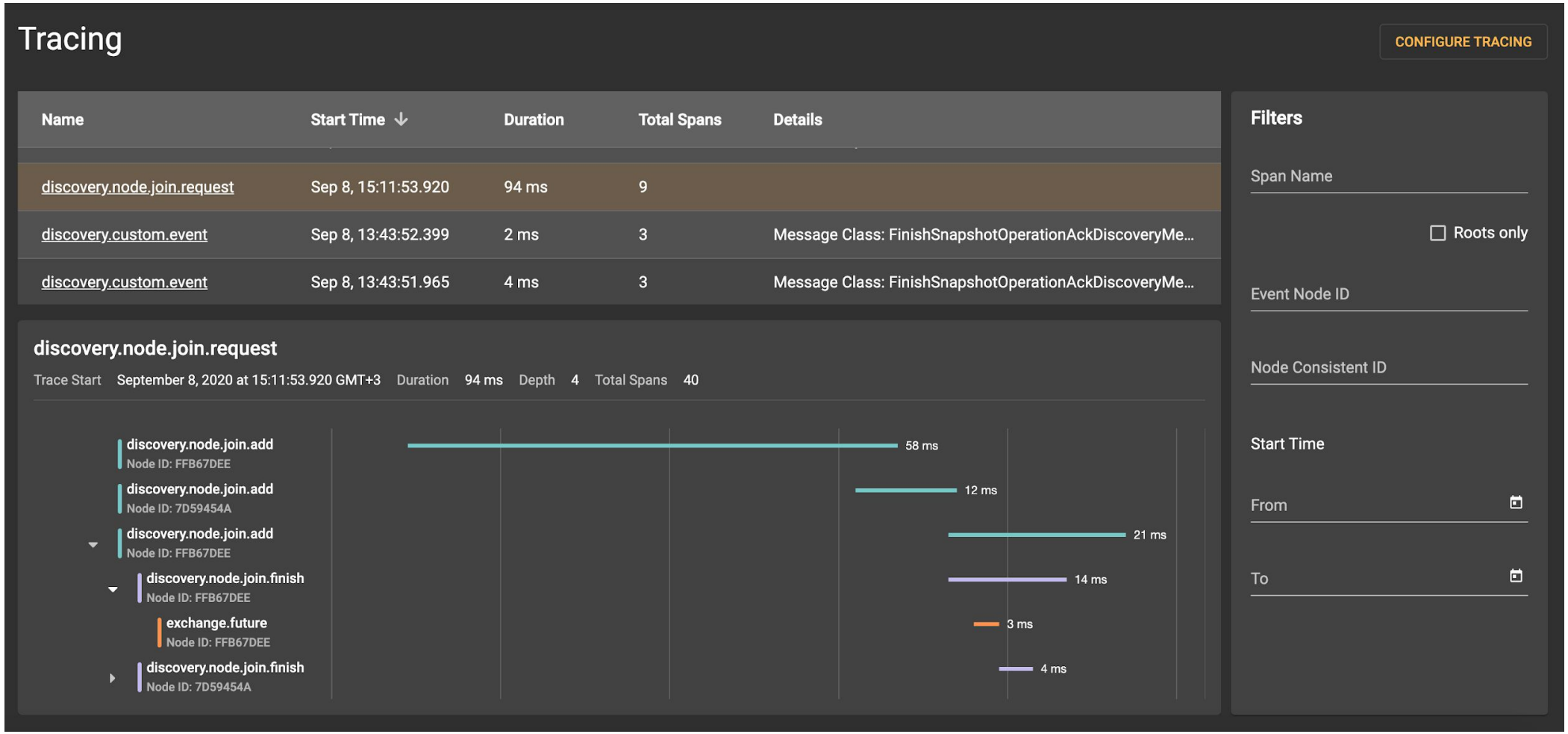
- Monitoring and management
- Hosted, on-premise
- Successor of Web Console

https://control.gridgain.com

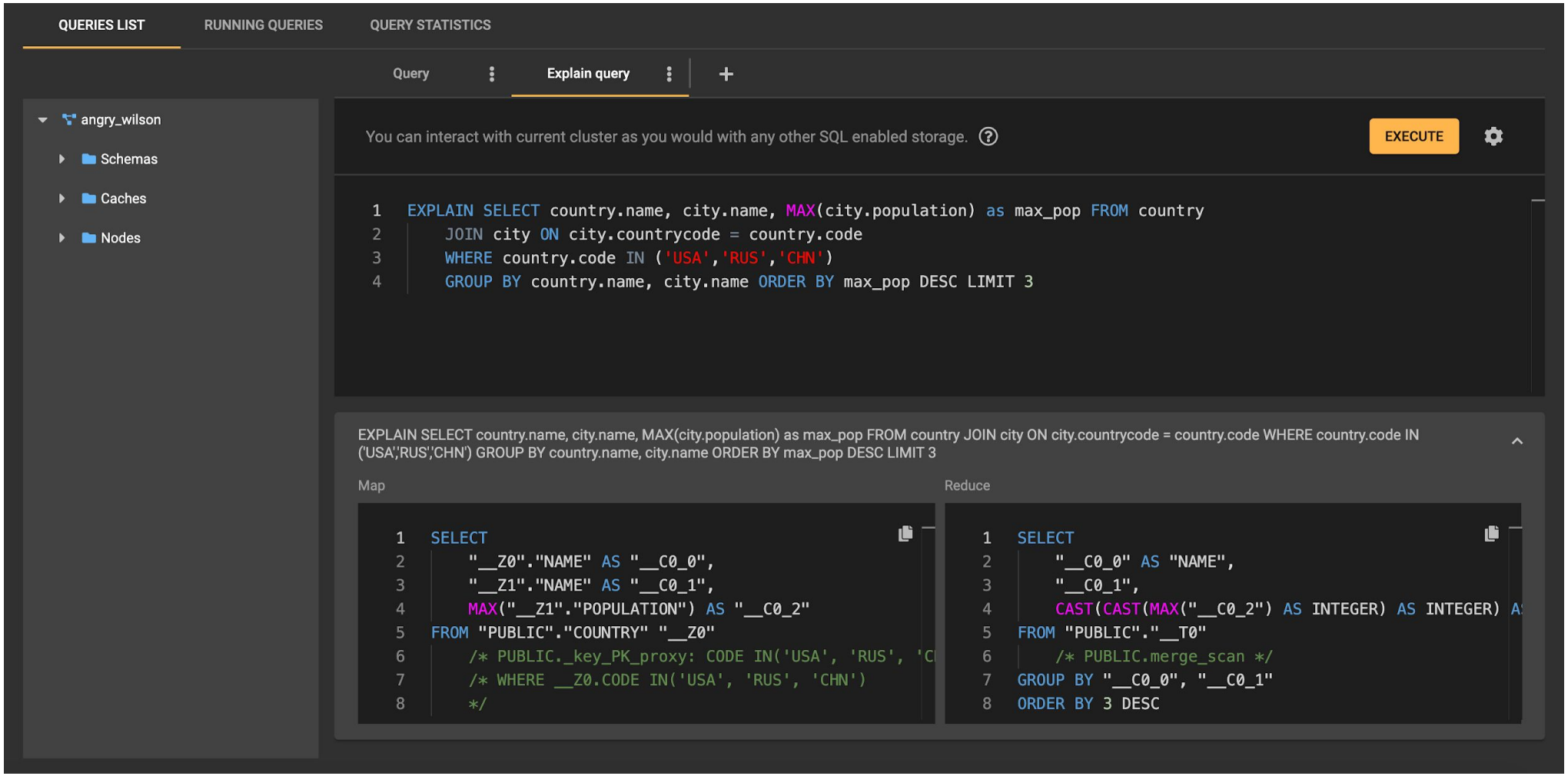
Control Center overview



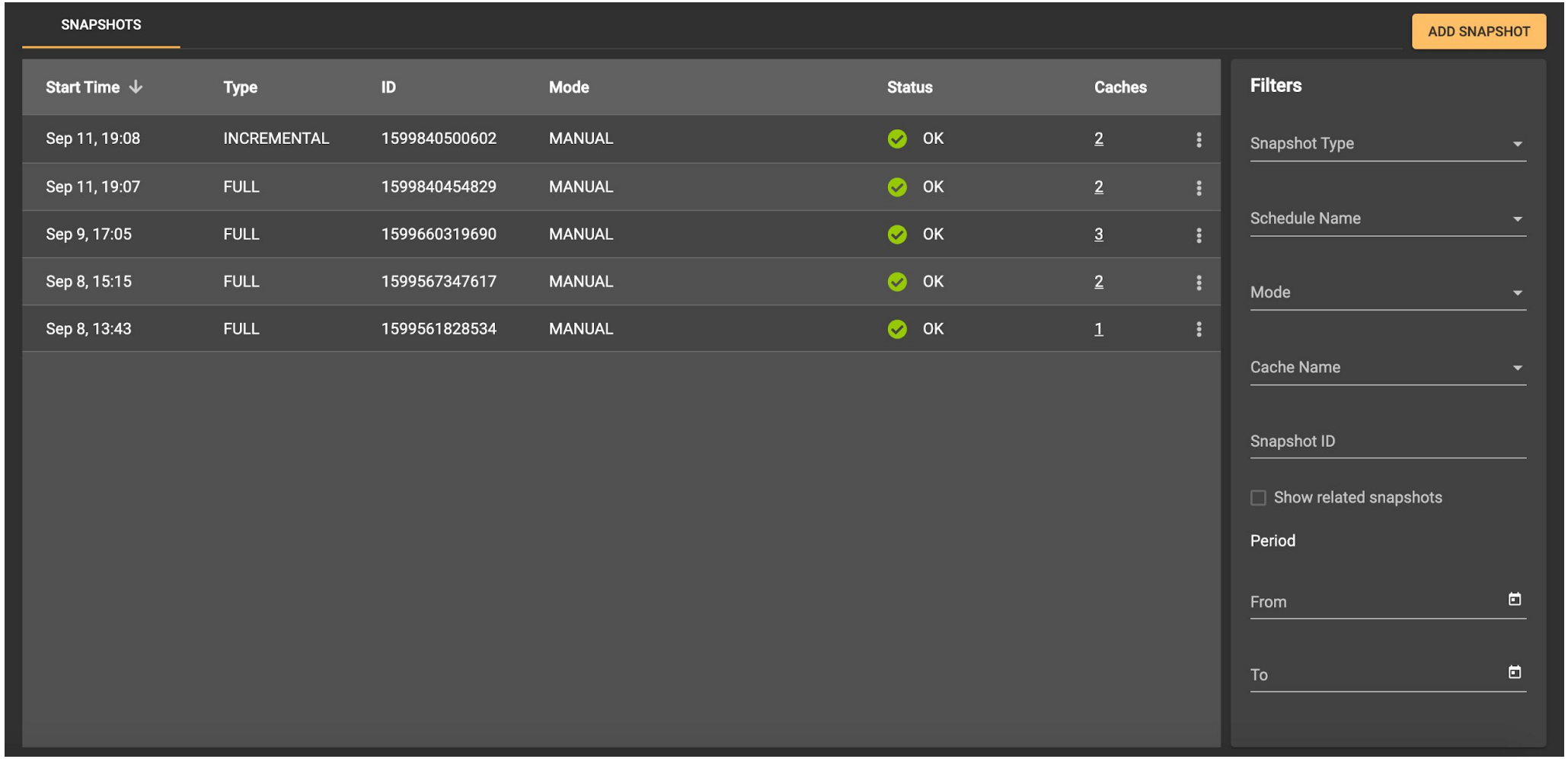
Control Center overview



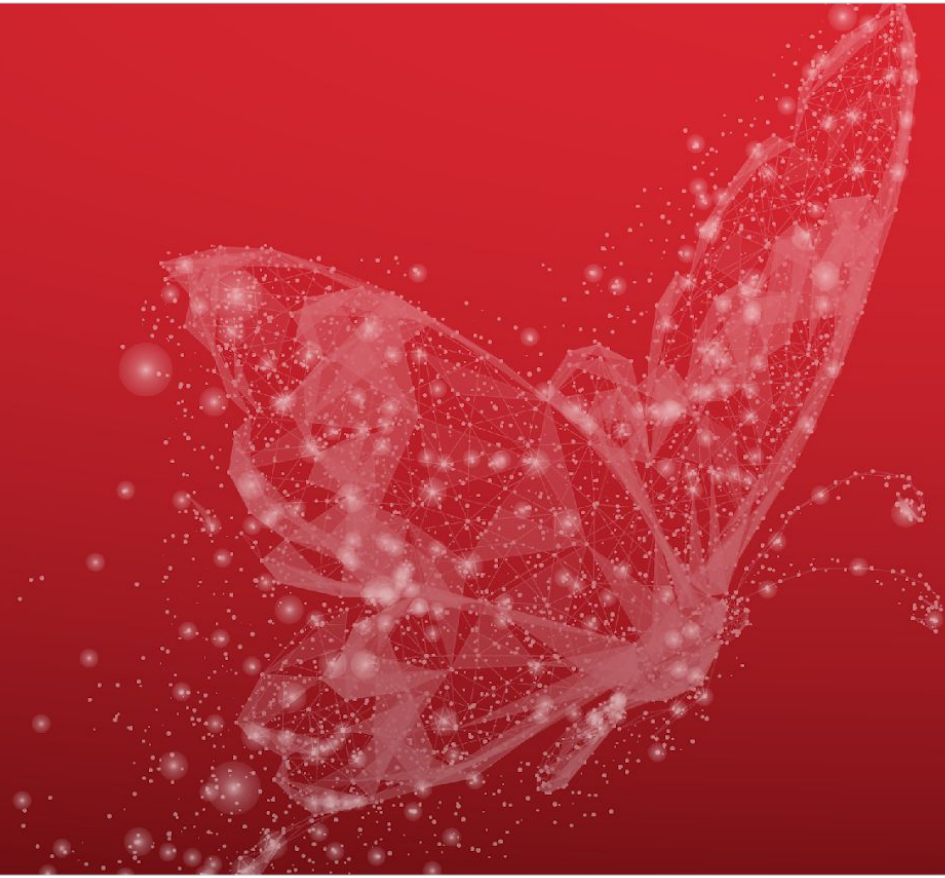
Control Center overview



Control Center overview

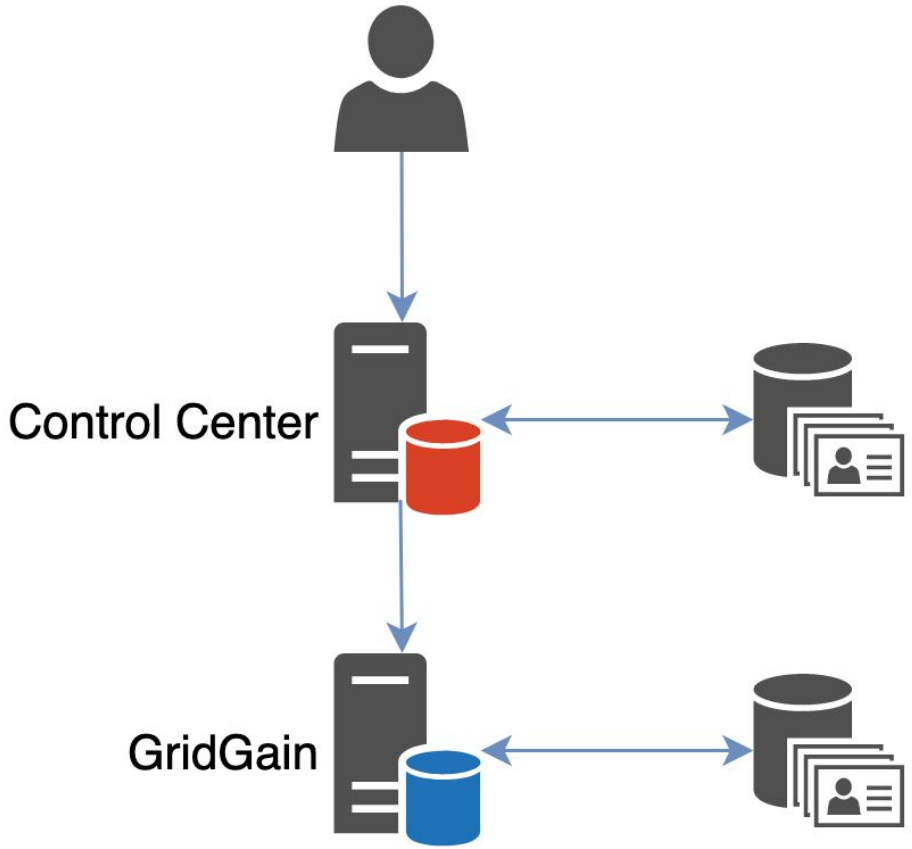


Demo. Control Center

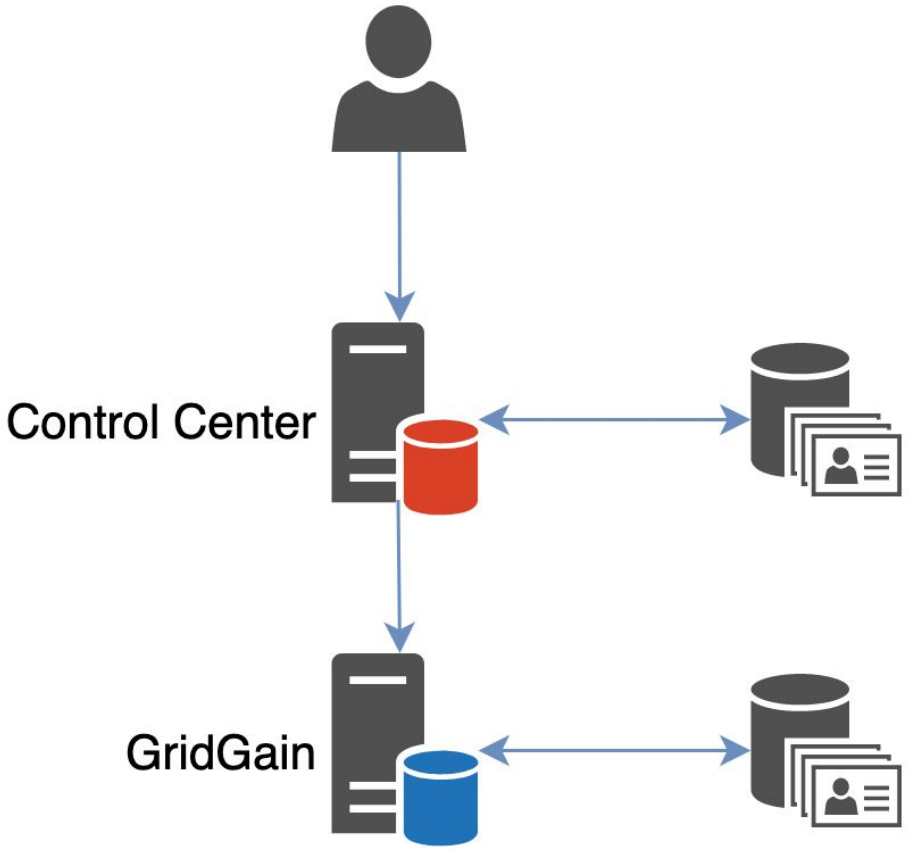


GridGain

Demo #1:
Separate users storage

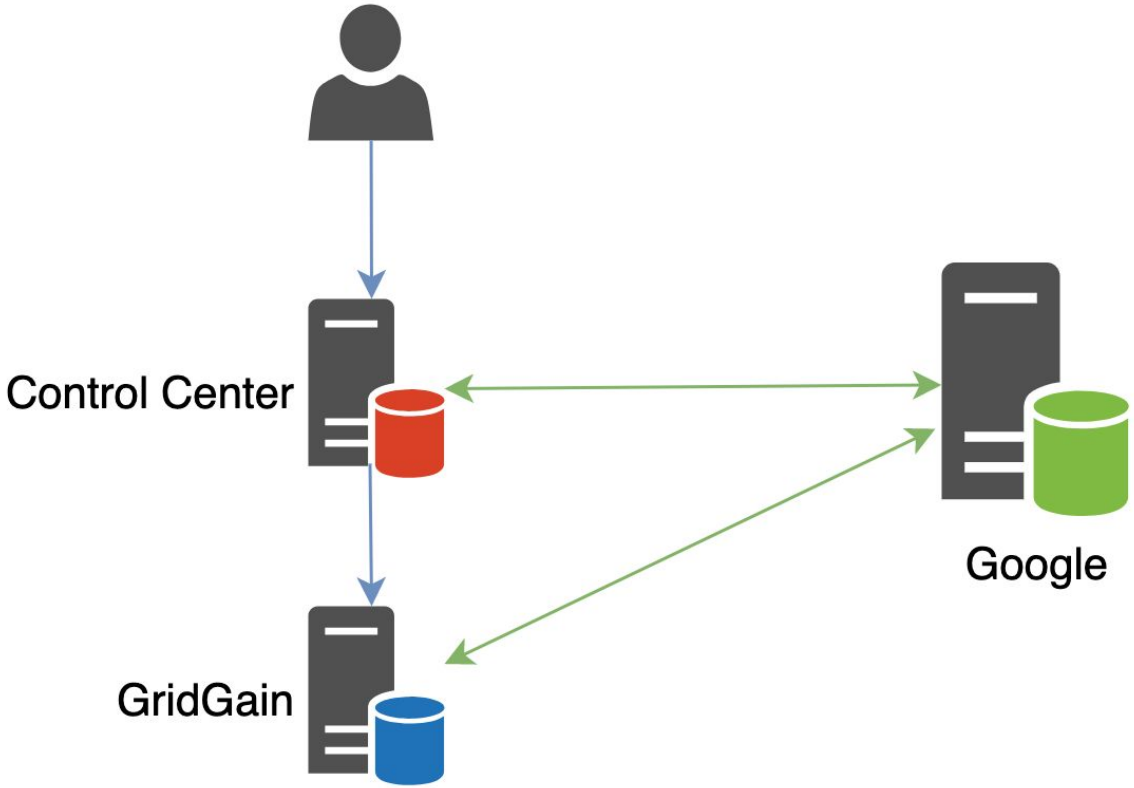


Demo #1:
Separate users storage

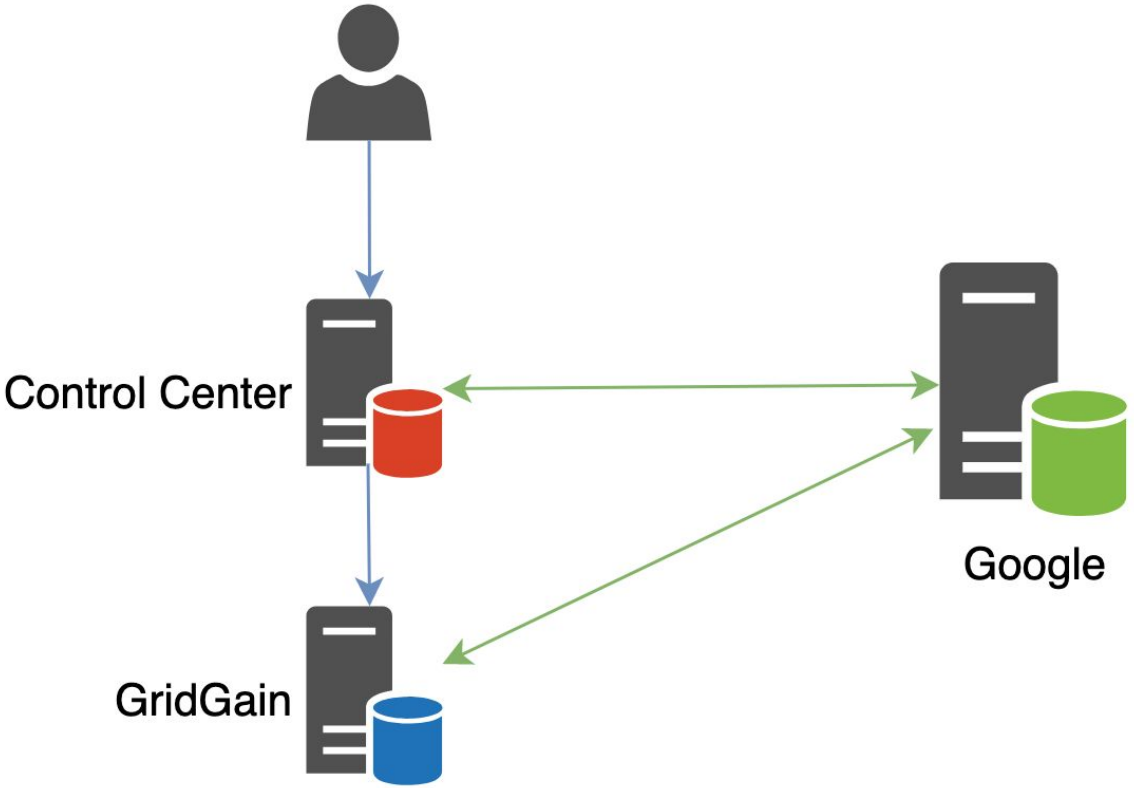


- ! Control Center and GridGain have different sets of users
- ! Repetitive password requests

Demo #2:
Centralized users storage

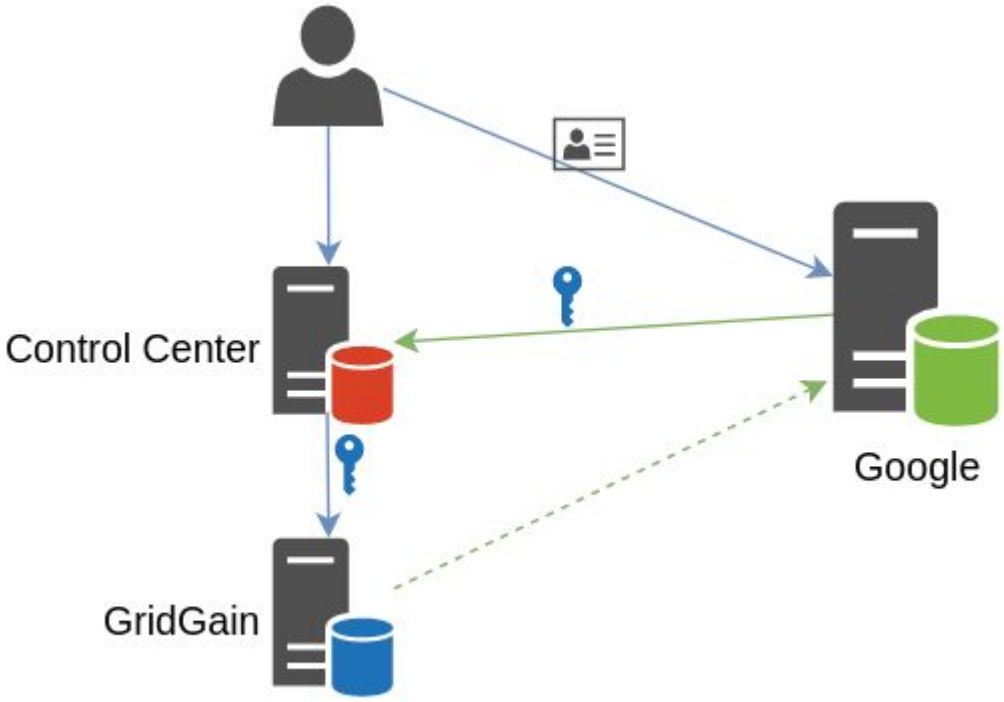


Demo #2:
Centralized users storage

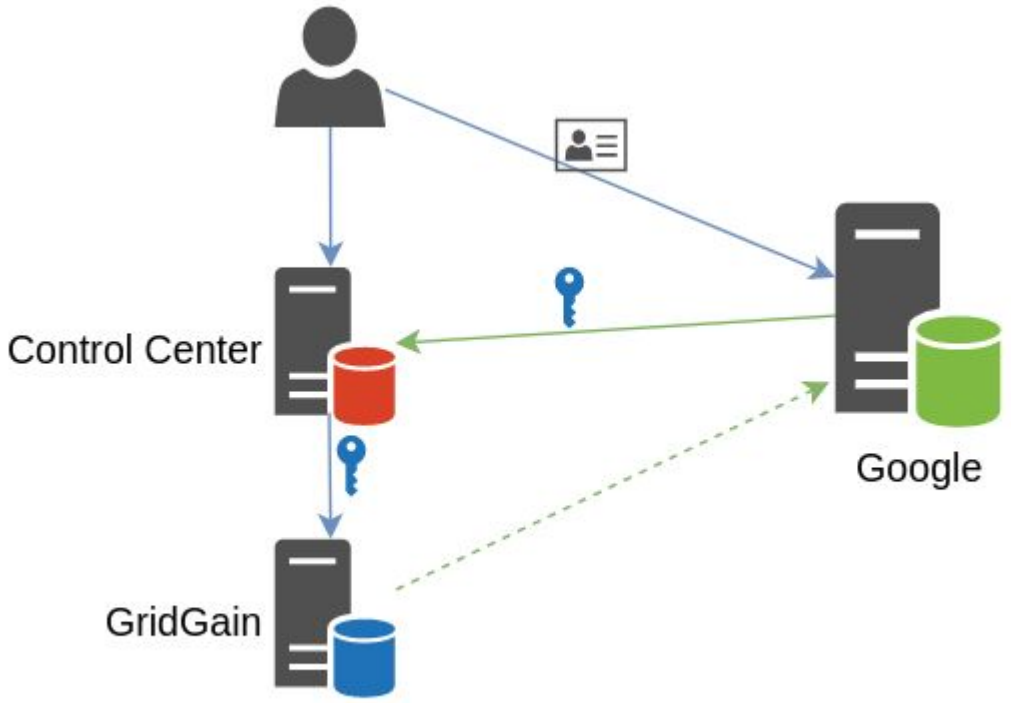


- The same set of users in both systems
- ! Repetitive password requests

Demo #3:
Single sign on

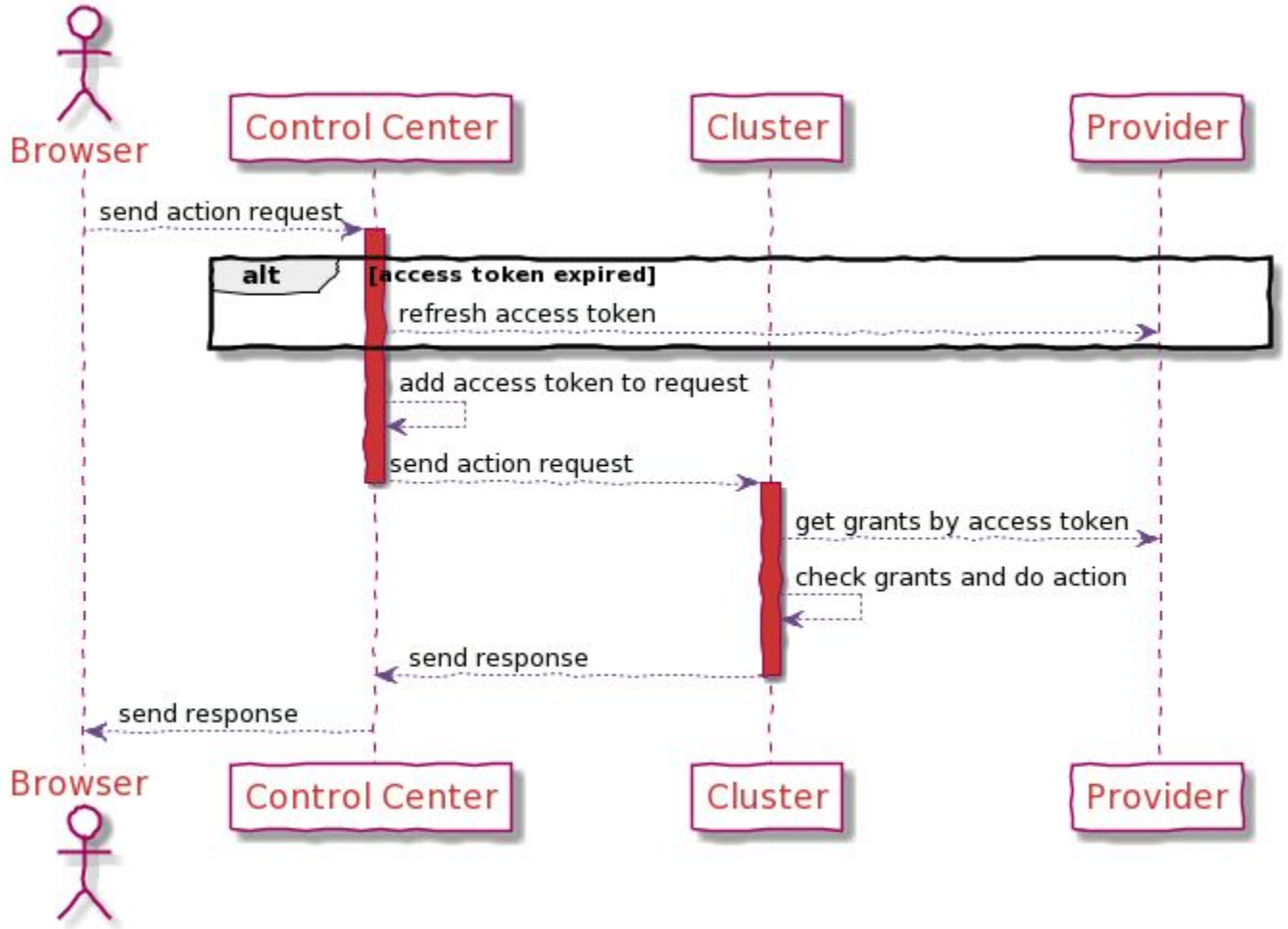


Demo #3:
Single sign on

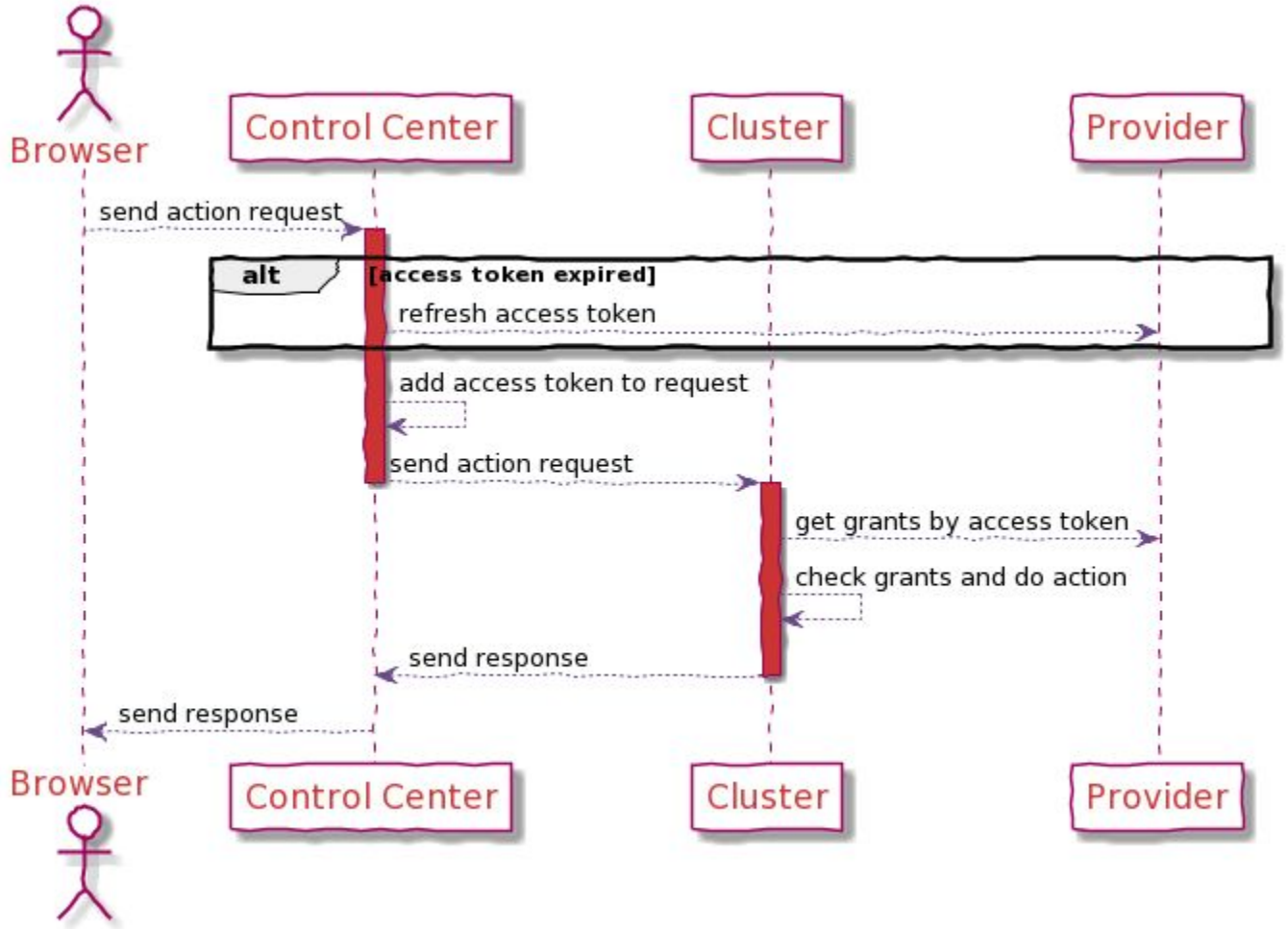


- The same set of users in both systems
- No need to type sign in multiple times

Single sign on flow

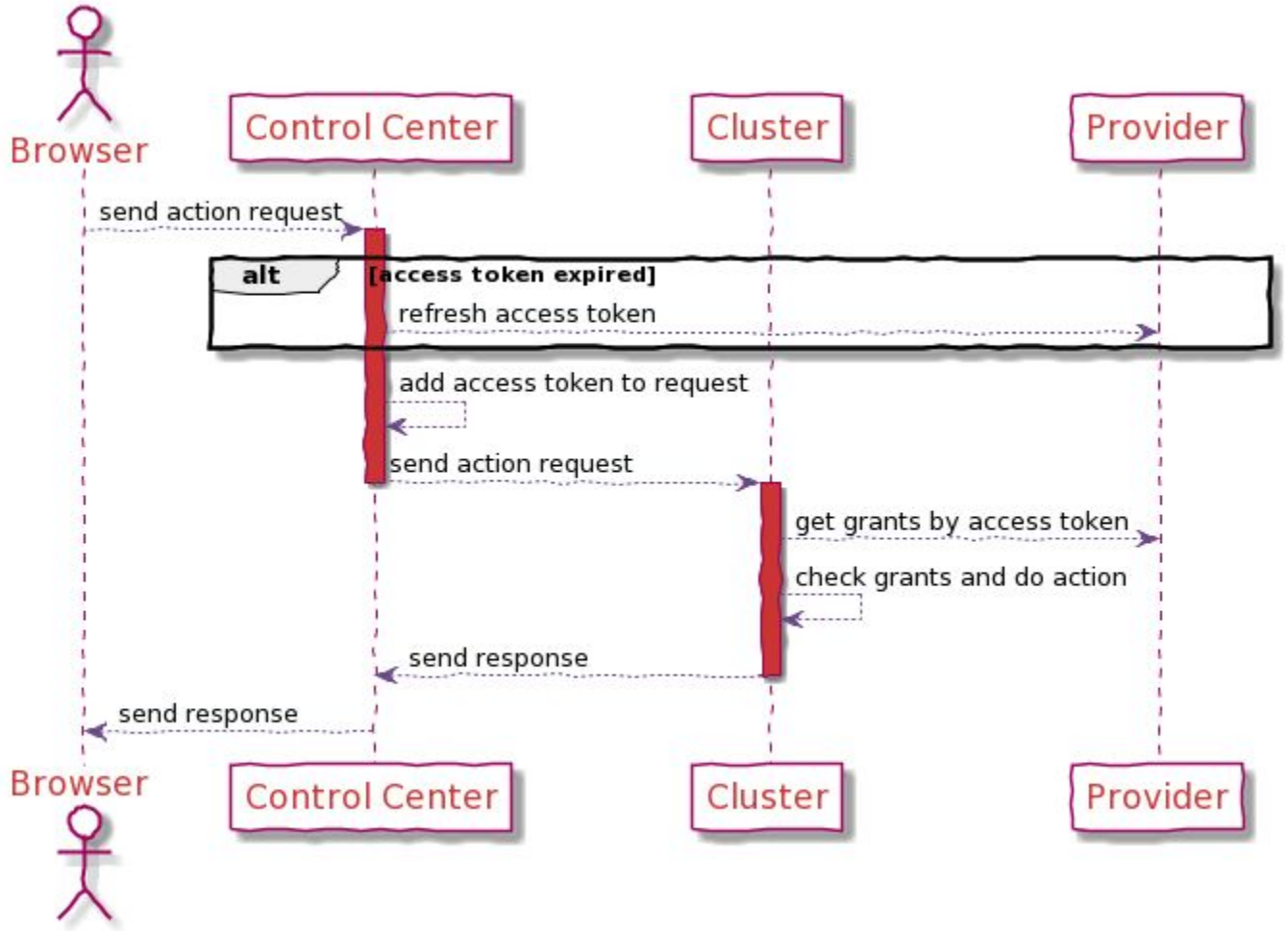


Single sign on flow




```
{
 "login": "firstuser",
 "password": "m5faSDf3",
 "userObject": null
}
```

Single sign on flow




```
{
 "login": null,
 "password": null,
 "userObject": {
   "tokenType": "Bearer",
   "accessToken":"SlAV32hkKG"
 }
}
```

GridGain Authenticator

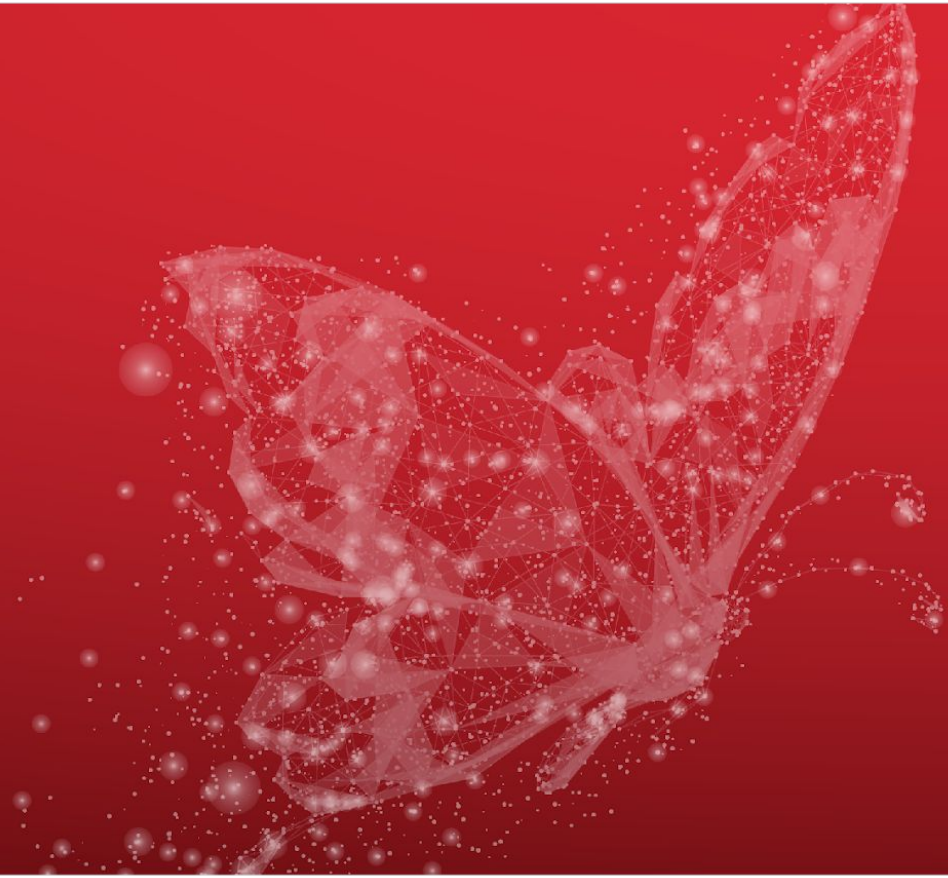
- Responsible for users authentication in the cluster
- Possible to extend to support authorization
- Available in GridGain Enterprise+

https://github.com/GridGain-Demos/gridgain-oauth2-authenticator-example/

OpenID Connect. Required steps.

- Register your application at the provider
  - Specify the redirect URL
- Specify the properties:
  - Client id
  - Client secret
  - Authorization URI
  - Token URI
  - JWK set URI

Demo
Adding another provider



Single sign on

! The setup is trickier

- Users management is more convenient
- The same credentials for the cluster and Control Center
- No repetitive identity checks
- Sessions are stored on the provider’s side

Control Center plans

- OAuth2 Authenticator
- More providers
- LDAP

Thank you!

Denis Mekhanikov
Team Lead @ GridGain Cloud Team
dmekhanikov@gridgain.com

Alexander Kozhenkov
Software Engineer @ GridGain Cloud Team
akozhenkov@gridgain.com