Training: Securing Service Provider Resources with Advanced Access Management

Valeria Ardizzone (EGI), Ignacio Lamata Martinez (EGI),
Nicolas Liampotis (GRNET)

19 June 2023

EGI Conference 2023

TLP: WHITE Public

How to secure access to your service with Check-in

Supported Authentication (AuthN) & Authorisation (AuthZ) Standards

- EGI Check-in can secure access to services using different protocols:
  - OAuth 2.0 / OpenID Connect (OIDC)
  - SAML (Security Assertion Markup Language)
  - X.509 Certificates

How to secure access to your service with Check-in

Integration Workflow

- To integrate your service with Check-in, you need to submit a registration request through the Federation Registry
- The integration follows a two-step process:

Step 1. Register your service and test integration with the **Demo** instance of EGI Check-in

- *Note: Register with the **Development** instance to test new features of Check-in that are not available in production yet*

Step 2. Register your service with the **Production** instance of EGI Check-in to enable Check-in users in production to access your service

How to secure access to your service with Check-in

Integration Environments

DEMO

- Used for Production-ready Services
- Supports all academic & social login options
- Service registration / reconfiguration requires approval by Operators:
  - Review of technical parameters

DEVELOPMENT

- Used for testing experimental features
- Supports limited login options
- Service registration / reconfiguration can be self-approved by service owners

PRODUCTION

- Used for Services in Production
- Supports all academic & social login options
- Service registration / reconfiguration requires approval:
  - Review of technical parameters
  - Review of policy aspects

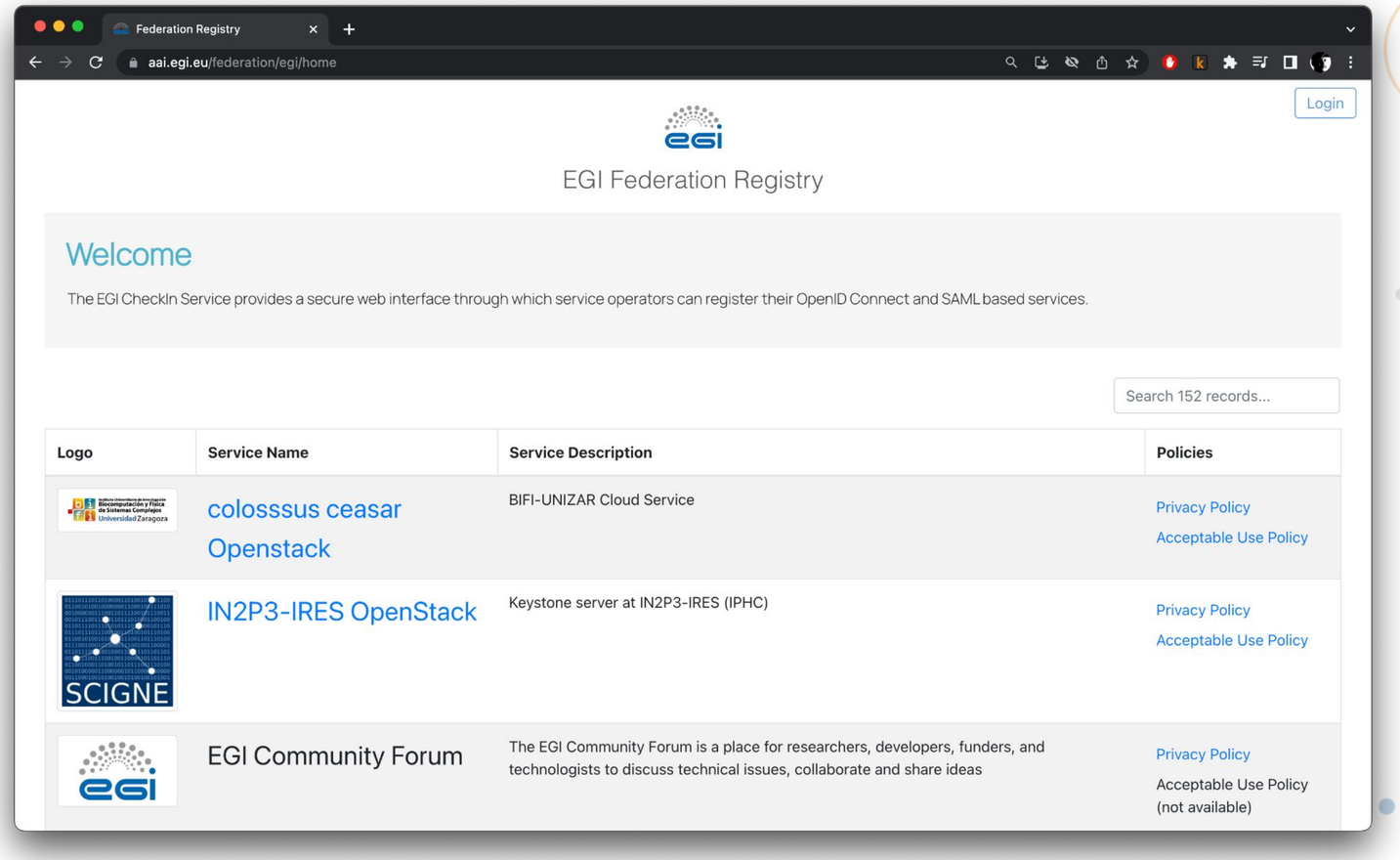
Federation Registry

How do I register my service?

How to secure access to your service with Check-in

Introduction to the Federation Registry

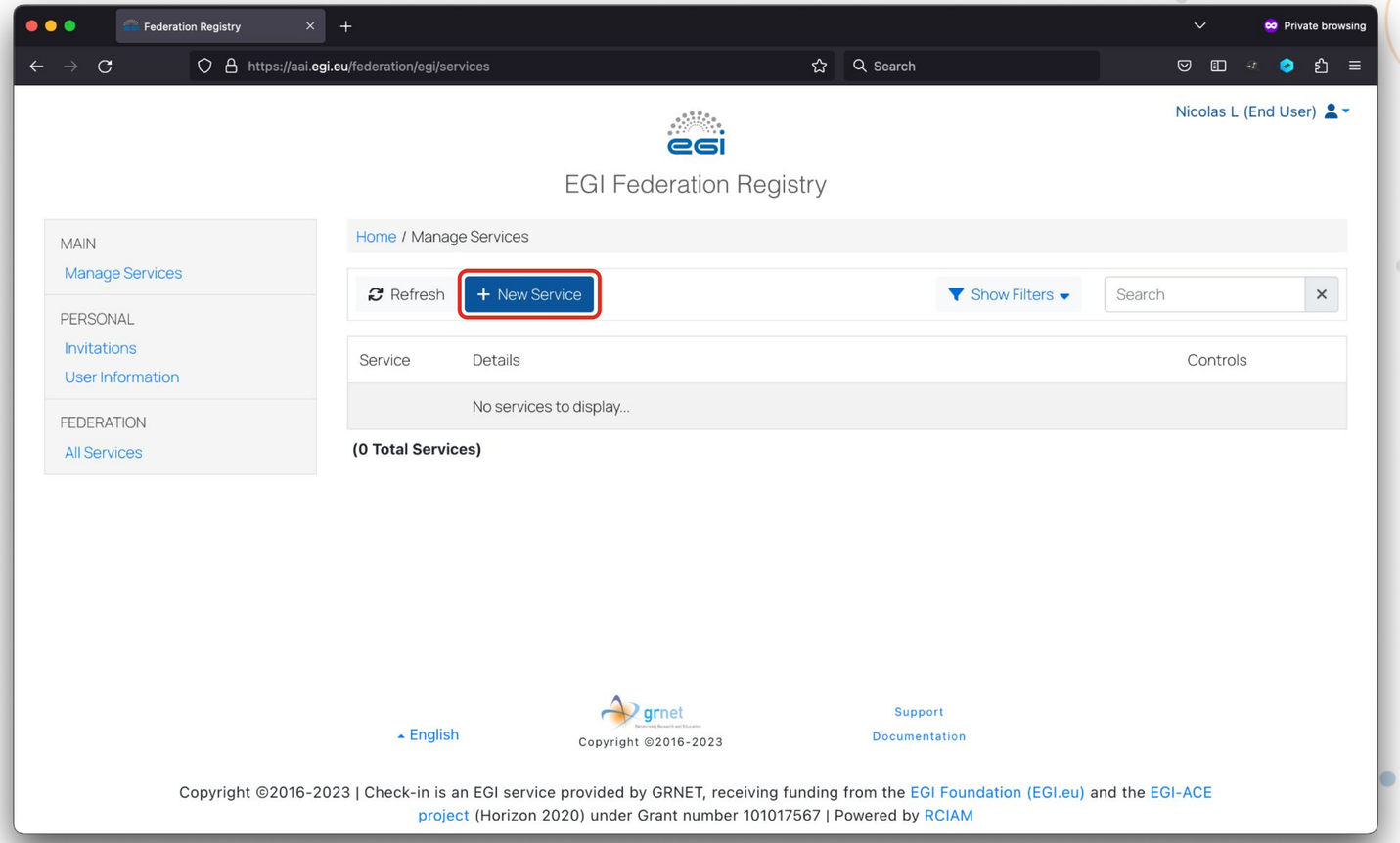
- Enables the secure registration, reconfiguration, and deregistration of service providers
- Provides single access point for managing services regardless of the underlying protocol (OpenID Connect vs SAML)
- Supports different AAI proxy technologies (Keycloak, SimpleSAMLphp, MITREid Connect)



Federation Registry

General guidelines

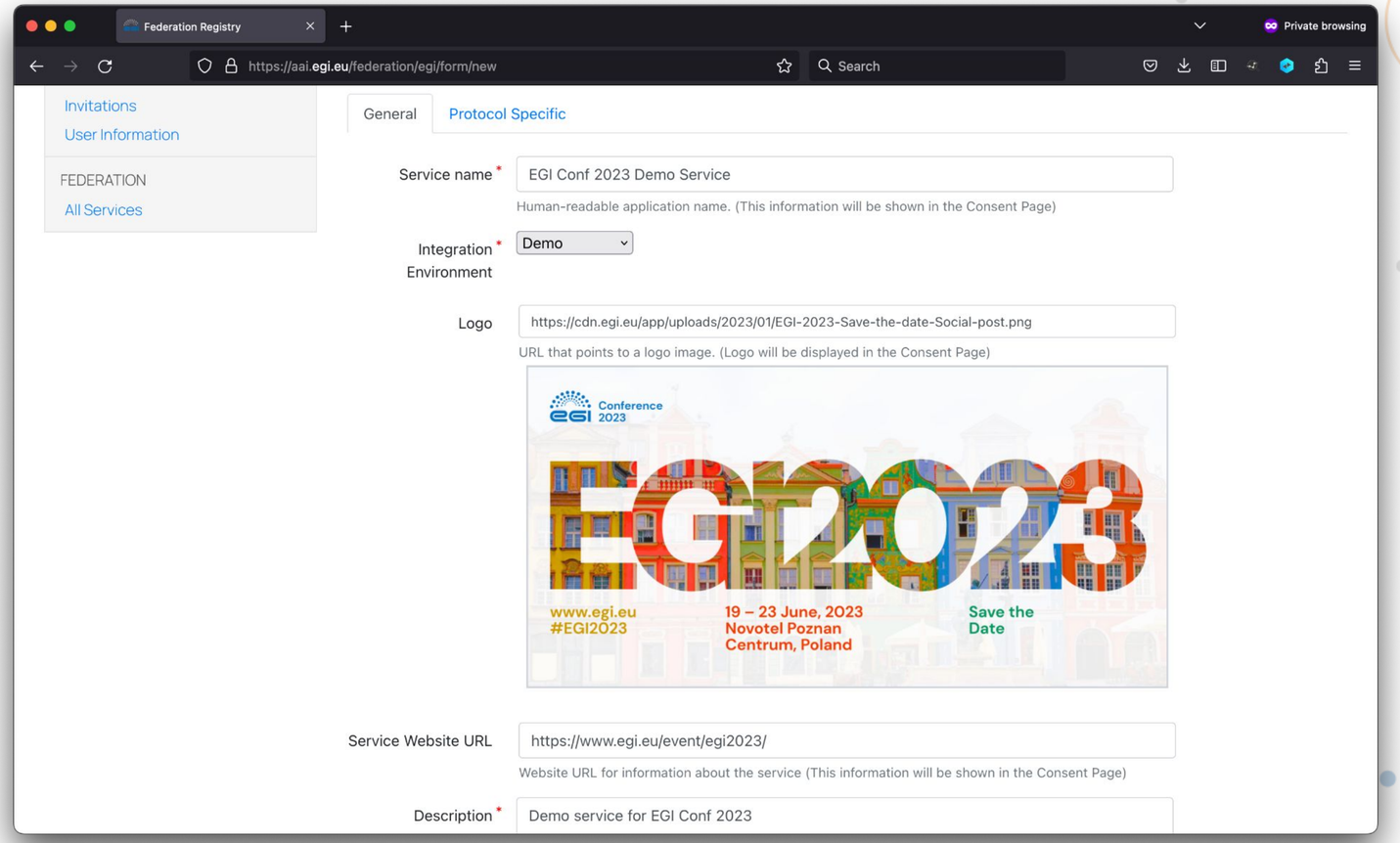
- Log into https://aai.egi.eu/federation
- Select **“Manage Services”** → **“New Service”**



Federation Registry

General guidelines

- In the **“General”** tab fill in the required information about your Service.
  - Select **“Demo”** as the “Integration Environment”.
- In the **“Protocol Specific”** tab configure the Service options (more info later on).
- Select **“Submit”**
- Wait for review process
- Test!



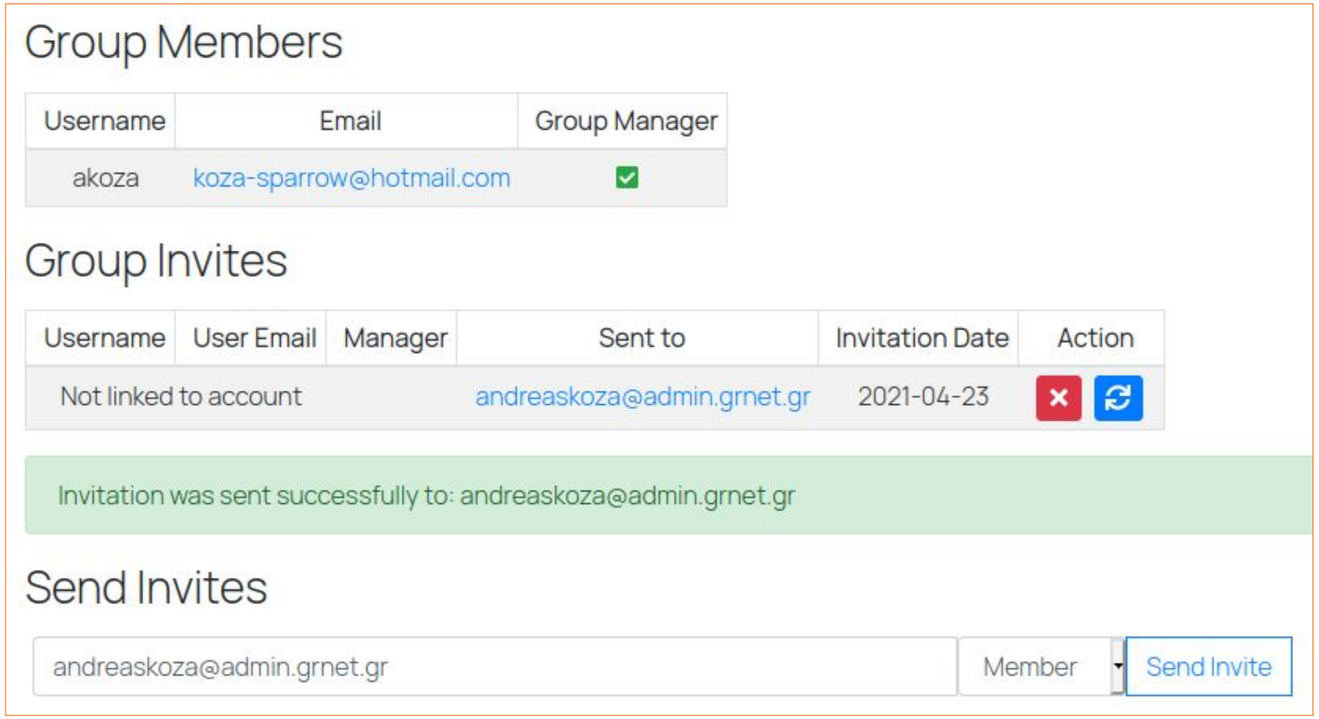
Federation Registry

We're a team, how can we manage our service collaboratively?

Federation Registry

Managing Service Owners

- A service is owned by a group of users called **owners group**
- Users in an owners group can take one of two roles:
  - **Member**
  - **Manager**
- Both **Members** & **Managers** can view and create reconfiguration requests for that service.
- **Managers** have control over group members, and can invite or remove users



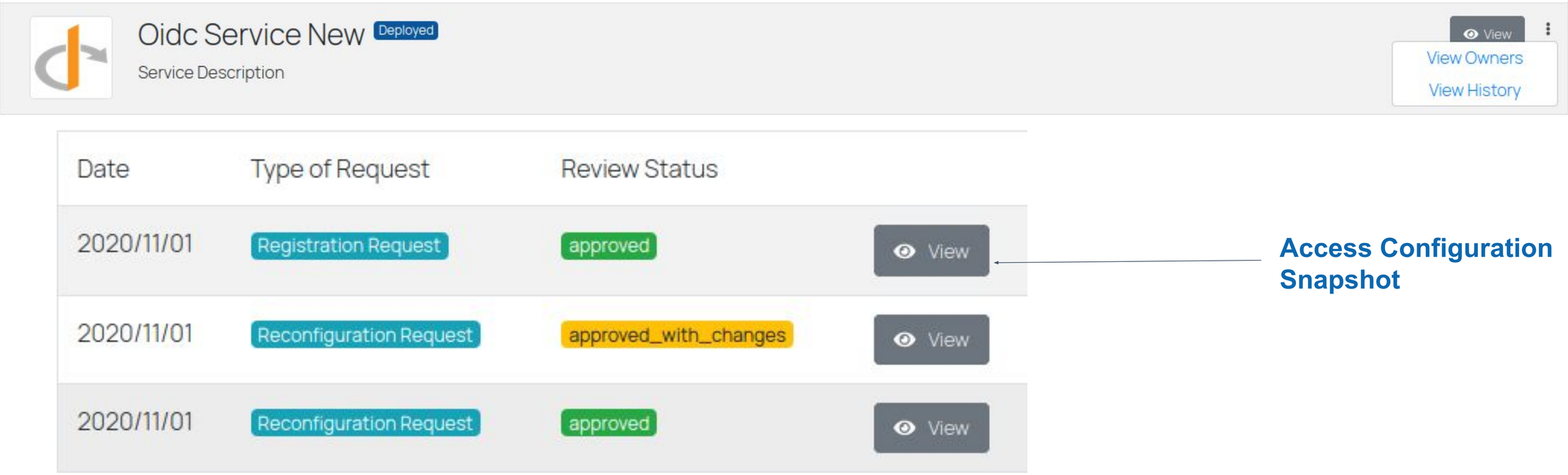
Federation Registry

How can I view the history of my service configurations?

Federation Registry

Service configuration history

- Service owners can view the History of their services
- Each time a petition is reviewed a new snapshot of the service configuration is added to History



Access Configuration Snapshot

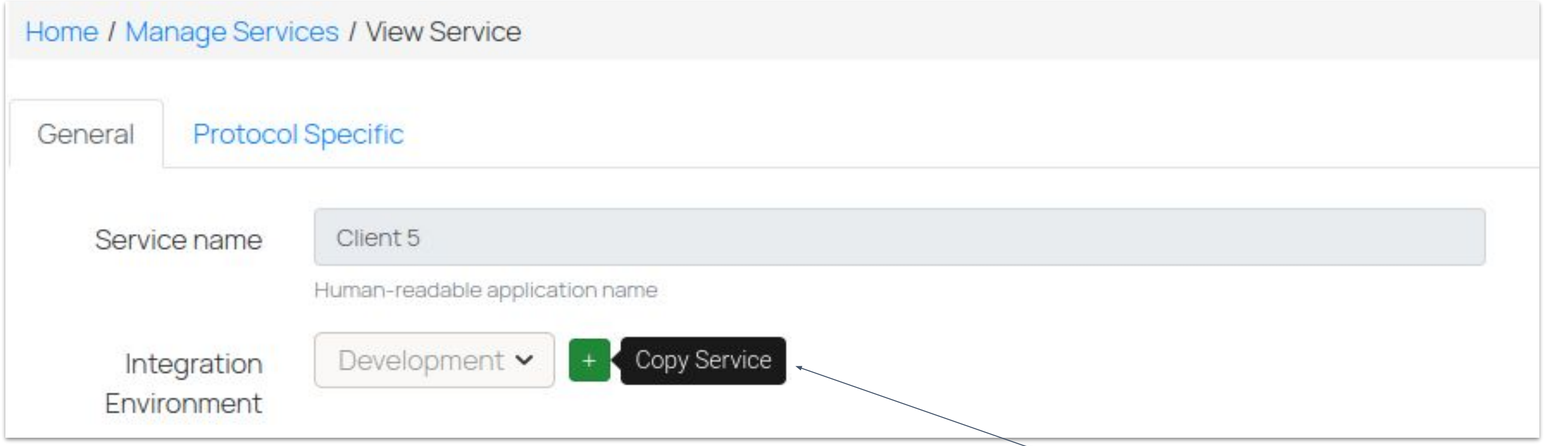
Federation Registry

How to enable my service in production?

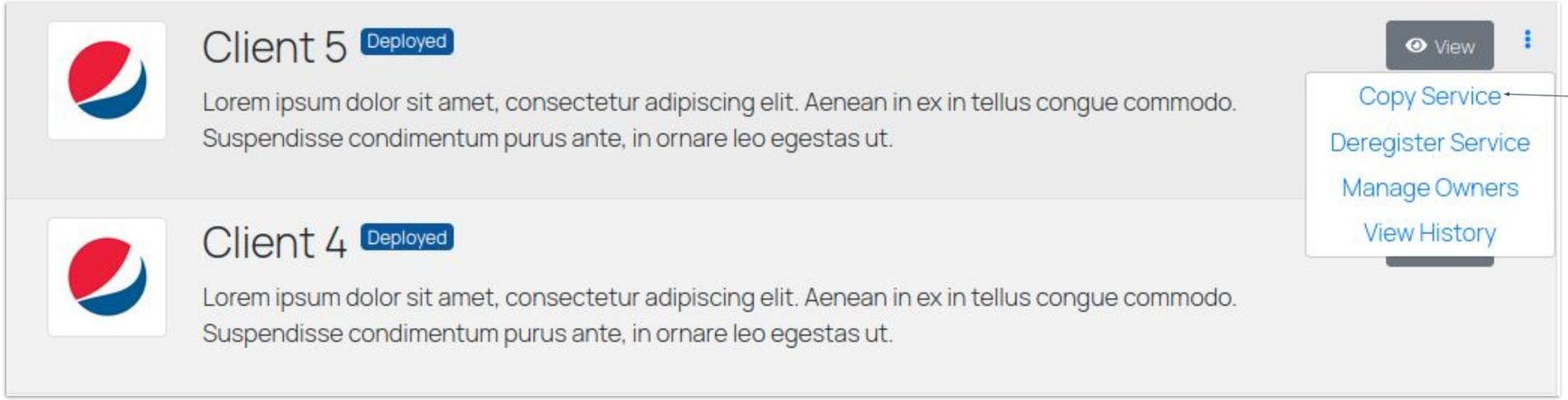
Federation Registry

Enabling service in production

- Use **"Copy Service"** option to copy the service configuration from current testing Integration Environment (typically "Demo") to "Production"
- Update the endpoints configured in your application accordingly
  - E.g. `https://aai-demo.egi.eu/…` → `https://aai.egi.eu/…`



When viewing the service



More Service Options Dropdown Menu

Authentication works! How do I control access?

How to secure access to your service with Check-in

Authorisation

1. Attribute-based authorisation
   - VO/Group membership and role information
   - GOCDB roles
   - Identity Assurance information
   - Affiliation with home organisation

2. Capability-based authorisation
   - Resources a user is allowed to access
   - Optional list of specific actions the user is entitled to perform

Attribute-based Authorisation

VO/Group membership and role information

- Allows services to control access to resources based on information about the VO/groups a user is a member of

- One or more values encapsulated in:
  - `eduPersonEntitlement` attribute (SAML)
  - `eduperson_entitlement` claim (OIDC)

- Each value formatted as a URN → AARC-G002

```
<NAMESPACE>:group:<VO>[:<GROUP>*][:role=<ROLE>]#<GROUP-AUTHORITY>
```

Attribute-based Authorisation

VO/Group membership and role information

- Example values:

urn:mace:egi.eu:group:vo.example.eu:role=member#aai.egi.eu

NAMESPACE VO ROLE GROUP-AUTHORITY

urn:mace:egi.eu:group:vo.example.eu:role=vm_operator#aai.egi.eu

NAMESPACE VO ROLE GROUP-AUTHORITY

urn:mace:egi.eu:group:vo.example.eu:cloud:role=member#aai.egi.eu

NAMESPACE VO GROUP ROLE GROUP-AUTHORITY

Attribute-based Authorisation

Assurance

Based on the authentication method selected by the user, EGI Check-in Proxy assigns a Level of Assurance (LoA):

- **Low**: Authentication through a social identity provider or other low identity assurance provider: `https://aai.egi.eu/LoA#Low`
- **Substantial**: Password/X.509 authentication at the user's home IdP: `https://aai.egi.eu/LoA#Substantial`

Capability-based Authorisation

- Capabilities can be used to convey authorisation information to services *in a compact form*
- One or more values encapsulated in:
  - `eduPersonEntitlement` attribute (SAML)
  - `eduperson_entitlement` claim (OIDC)
- URN syntax → AARC-G027

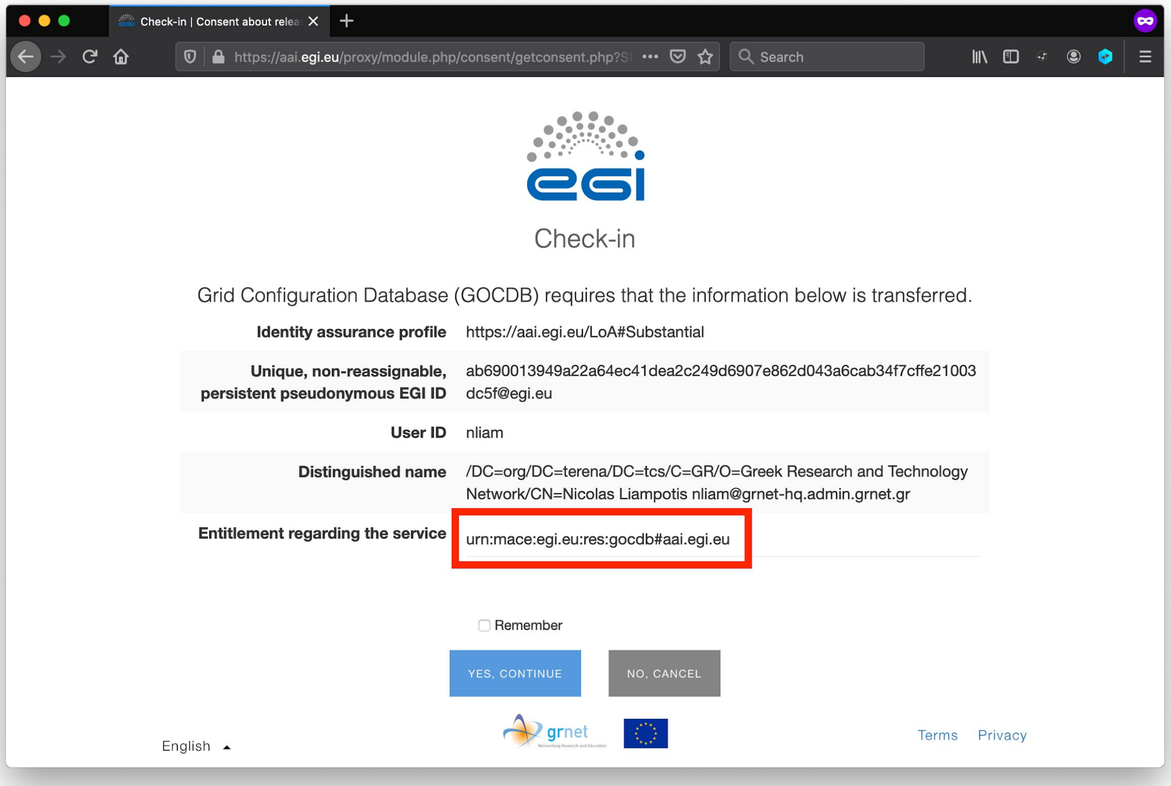
```
<NAMESPACE>:res:<RESOURCE>[:<CHILD-RESOURCE>]...
[:act:<ACTION>[,<ACTION>]...]#<AUTHORITY>
```

Capability-based Authorisation

- Example value:

`urn:mace:egi.eu:res:gocdb#aai.egi.eu`

NAMESPACE RESOURCE AUTHORITY



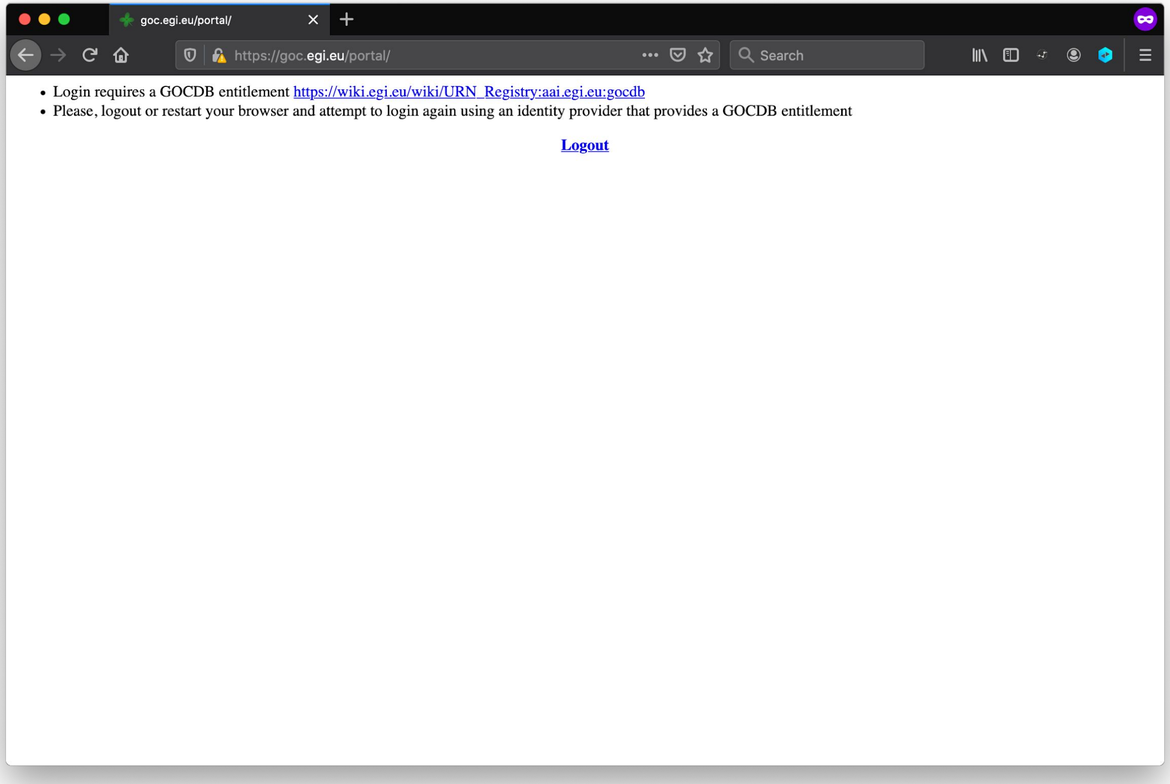
Capability-based Authorisation

- Example value:

`urn:mace:egi.eu:res:gocdb#aai.egi.eu`

NAMESPACE RESOURCE AUTHORITY

So what if I don't have a capability?



OIDC & OAuth 2.0 in action!

How to secure access to your service with Check-in

Testing OIDC/OAuth 2.0 flows

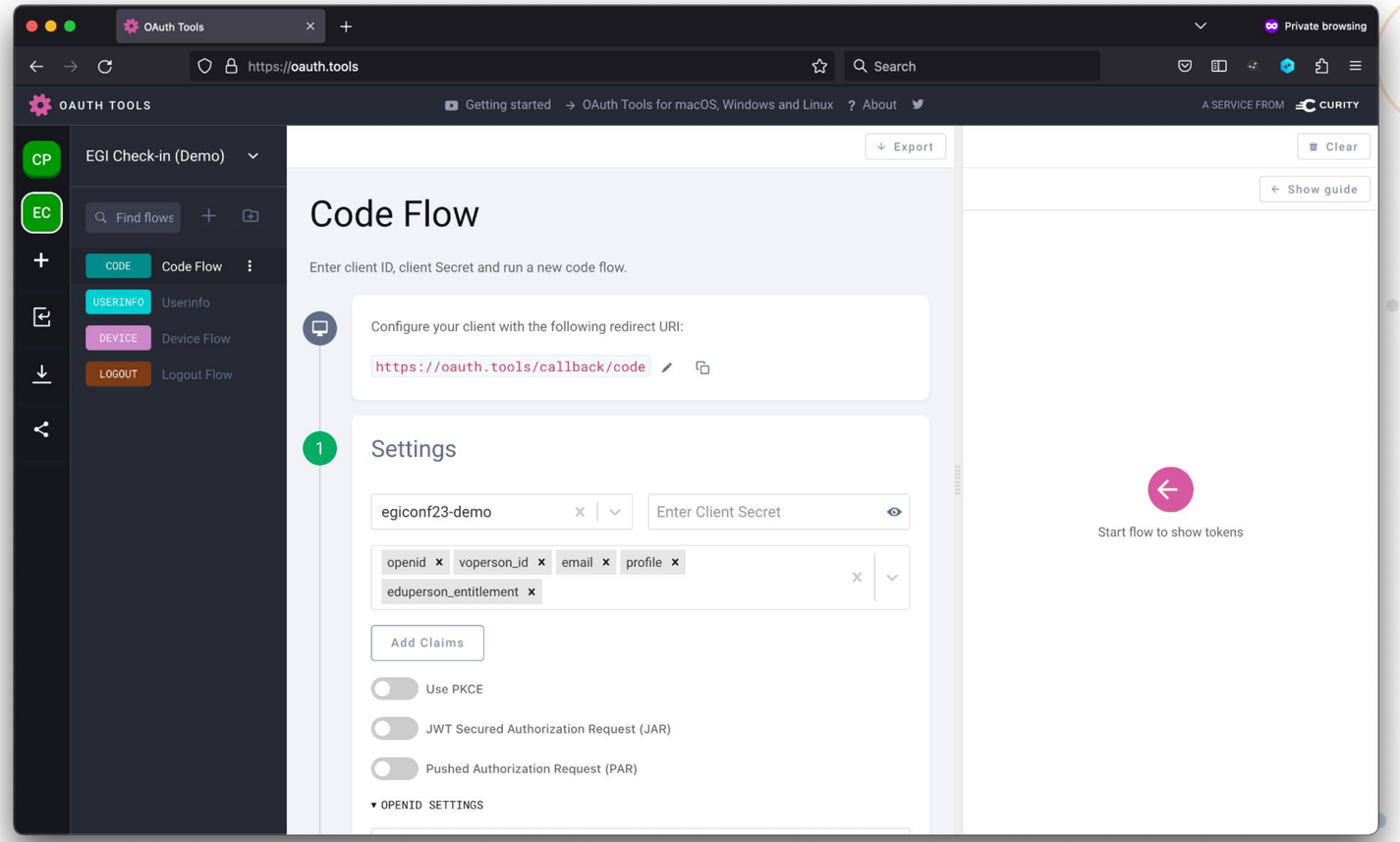
- Download oauth tools config:

https://go.egi.eu/oauthconfig

How to secure access to your service with Check-in

Testing OIDC/OAuth 2.0 flows

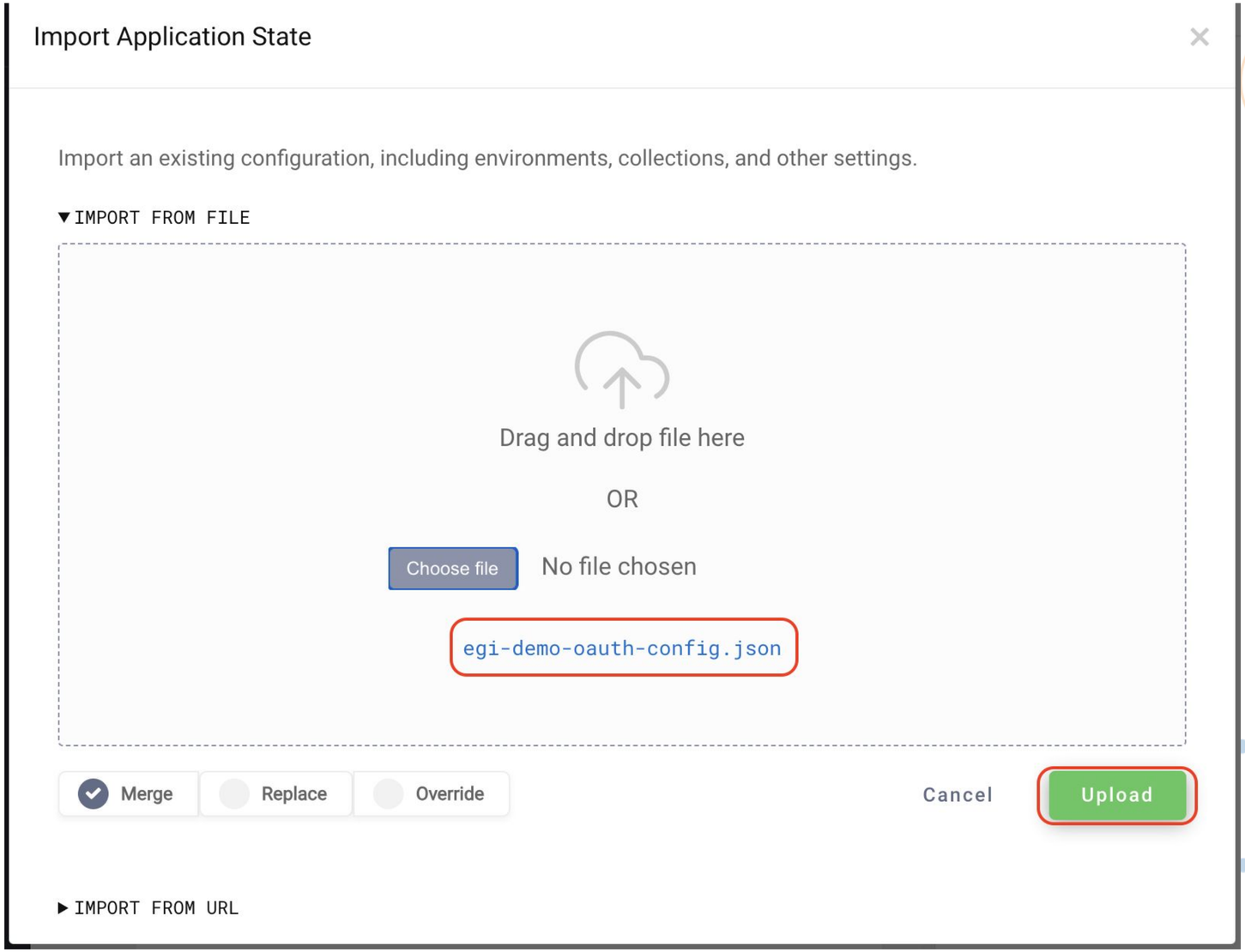
- Go to https://oauth.tools
- Select **“Import”** option from side menu and select **`egi-demo-oauth-config.json`**



How to secure access to your service with Check-in

Testing OIDC/OAuth 2.0 flows

- Go to https://oauth.tools
- Select **“Import”** option from side menu and select **`egi-demo-oauth-config.json`**
- Select **“Upload”**



How to secure access to your service with Check-in

OIDC/OAuth 2.0 flows

- Public client using AuthZ Code Grant with Proof Key for Code Exchange (PKCE)
- Device AuthZ Grant
- Retrieve User Profile information from UserInfo endpoint
- Resource Server using Token Introspection

OIDC/OAuth 2.0 flows

Public client using AuthZ Code Grant with Proof Key for Code Exchange (PKCE)

- Why PKCE?
  - Public clients (i.e. no Client Secret) **using** the Authorization Code Grant are susceptible to the authori**s**ation code interception attack
- Use cases:
  - Single-page applications
    - Source is available to the browser → Cannot securely store a Client Secret
  - Native applications:
    - Decompiling the app will reveal the Client Secret → Cannot securely store a Client Secret
    - Custom URL scheme for redirects → Malicious applications can receive an Authorization Code from the Authorization Server.

Is my service ready for production?

How to secure access to your service with Check-in

Checklist for production

- User attributes
  - User identifier → `voperson_id` scope/claim or `voPersonID` SAML attribute?
  - Are these attributes described in your privacy policy?
- Authorisation:
  - Thoroughly test Attribute- or Capability based access control model
  - Error Handling for Access Denied
- User consent:
  - Check information (service name & description, Privacy policy & AUP, Support contact) shown to end-users
- Endpoints:
  - Check that protocol endpoints are properly protected with TLS / SSL certificates
- OIDC/OAuth 2 specific:
  - Does your client retrieve Check-in's OP configuration based on `issuer + /.well-known/openid-configuration` (see OpenID Connect Discovery)

Contact us

check-in@egi.eu

Let's talk. Or
meet in person

Get in touch with us

www.egi.eu

This work is partially funded by the EU research and innovation programme