The EGI AAI Check-In service for scientific communities

Nicolas Liampotis (GRNET)
Valeria Ardizzone (EGI Foundation)
Connect your Service to EGI Check-in
Connect your Service to EGI Check-in

General guidelines

To integrate your Service Provider with the EGI Check-in service, you need to submit a GGUS ticket indicating your request.

The integration follows a two-step process:

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

You can also test new features of Check-in that are not available in production yet, by registering your Service Provider and testing integration with the Development instance of Check-in.

**Step 2.** Register your Service Provider with the Production instance of EGI Check-in to allow members of the EGI User Community to access your service.
Connect your Service to EGI Check-in

General guidelines

To open a ticket go to GGUS and fill the following information about the Service:

• The name of the service
• A short description
• The logo of the service
• Contact information
  ▪ Support
  ▪ Technical
  ▪ Security
• The URL to the policy statement of the service
• Protocol-specific service configuration (e.g. SAML metadata or OIDC scopes & redirect URIs)
Connect your Service to EGI Check-in

General guidelines

<table>
<thead>
<tr>
<th>Issue information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date / Time of issue: 2020-05-27 14:00 UTC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>* Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register &lt;SAML SP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>* Describe the issue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hello Check-in AAI Support team, I would like to register my Service. Here is the service information:</td>
</tr>
<tr>
<td>- Name:</td>
</tr>
<tr>
<td>- Description:</td>
</tr>
<tr>
<td>- Logo URL:</td>
</tr>
<tr>
<td>- Contact info:</td>
</tr>
<tr>
<td>- Support:</td>
</tr>
<tr>
<td>- Technical:</td>
</tr>
<tr>
<td>- Security:</td>
</tr>
<tr>
<td>- Policy Statement:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Concerned VO?</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Affected site?</th>
</tr>
</thead>
<tbody>
<tr>
<td>please select</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>* Ticket category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Request</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type of issue?</th>
</tr>
</thead>
<tbody>
<tr>
<td>please select</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attach File(s) (max. 2 MB pro File)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse... No file selected.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Routing information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expert option, please set this option only if you know what it means.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notify SITE?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submit</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Assign to support unit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check-in (AAI)</td>
</tr>
</tbody>
</table>
My Service supports SAML, what should I do?
To enable federated access to a web-based application, you can connect to the EGI Check-in IdP as a SAML Service Provider (SP). Once the user is authenticated, the EGI Check-in Proxy will return a SAML assertion to the application containing information about the authenticated user.
SAML authentication relies on the use of metadata. Both parties (you as a SP and the EGI Check-in IdP) need to exchange metadata in order to know and trust each other. The metadata include information such as the location of the service endpoints that need to be invoked, as well as the certificates that will be used to sign SAML messages.

It is important that you serve your metadata over HTTPS using a browser-friendly SSL certificate, i.e. issued by a trusted certificate authority.
Metadata provided by your SP should contain:

- a descriptive name of the service
- email address for:
  - support
  - technical
Add the EGI Check-in IdP metadata to the trusted IdPs of your SP, which can be found in the following URL: https://aai.egi.eu/proxy/saml2/idp/metadata.php

Then include the metadata of your Service in the ticket
Register SAML Service

<table>
<thead>
<tr>
<th>Issue information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date / Time of Issue</td>
</tr>
<tr>
<td>Subject?</td>
</tr>
</tbody>
</table>
| Describe the issue? | Hello Check-in AAI Support team,
I would like to register my service.
Here is the service information:
- Name:  
- Description:  
- Logo URL:  
- Contact Info  
  - Support:  
  - Technical:  
  - Security  
- Policy Statement:  
- Metadata URL: |
My Service supports OIDC, what should I do?
Register OIDC Client

Login to https://aai-dev.egi.ei/oidc and create a new client
Register OIDC Client

In “Main” tab fill the following:

- Client name
- Redirect URI(s)
- Description
- Logo
- Policy Statement
- Contacts
In “Access” tab fill the following:

- Scope
- Grant Types
- Response Types
- Introspection
When the integration of the Service with development instance of EGI Check-in is completed create a [GGUS ticket](#) and request to move the client to production instance.
Register OIDC Client

Issue information

Date / Time of Issue: 2020-05-27 14:00 UTC

* Subject: Register OIDC Client to EGI Check-in production instance

* Describe the Issue:
Hello Check-in AAI Support team,

I would like to move my Client (client_id: XXX) to EGI Check-in production instance.
My Service has been integrated with EGI Check-in, how do I control user access?
Authorisation

1. Attribute-based authorisation
   - VO/Group membership and role information
   - GOCDB roles
   - 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 vs. Capability-based authorisation

The two models can co-exist even within the same service

- Capability-based authorization
- Attribute-based authorization

Slide courtesy of B. Bockelman
Attribute-based Authorization

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 Authorization

Group membership and role information

- Example values:

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

  - `urn:mace:egi.eu:group:vo.example.eu:role=vm_operator#aai.egi.eu`
Based on the authentication method selected by the user, EGI Check-in Proxy assigns a Level of Assurance (LoA).

EGI Check-in currently distinguishes between three LoA levels, similarly to the eID Assurance Framework (eIDAF). Each level is represented by a URI as follows:

- **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
- **High**: Substantial + multi-factor authn (not supported yet): https://aai.egi.eu/LoA#High
Attribute-based Authorisation

Assurance information - Coming soon...

<table>
<thead>
<tr>
<th>Should identifiers be unique, personal and traceable?</th>
<th>Should identifiers be unique across the infrastructure?</th>
<th>How fresh do attributes need to be?</th>
<th>What kind of ID Proofing is required?</th>
<th>Is Multi-Factor Authentication required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unspecified</td>
<td>Unspecified</td>
<td>Unspecified</td>
<td>Unspecified</td>
<td>Unspecified</td>
</tr>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>1 month</td>
<td>Low (self asserted)</td>
<td>Single factor authentication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medium (e.g. postal credential delivery)</td>
<td>Multifactor authentication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>High (e.g. face to face)</td>
<td></td>
</tr>
</tbody>
</table>

New profiles!

- AARC Assam
- IGTF Dogwood
- RAF Cappuccino
- IGTF Birch
- RAF Espresso
• 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**
Example value:

urn:mace:egi.eu:res:goCD#aai.egi.eu

- **NAMESPACE**
- **RESOURCE**
- **AUTHORITY**

So what if I don't have a capability?
Authentication & Authorisation

The big picture

- HO user identifier
- Name
- Email
- Affiliation
- Assurance

eduGAIN

Social

- Unique Check-in identifier → linked to one or more HO identities
- Assurance
- VO/group membership & roles
- GOCDB roles
- Capabilities

Service

EDGAIN CheckIn

Virtual Organization

EOSC Infrastructure
Live Demo

Authorization Code Flow
The authorization code is a temporary code that the client will exchange for an access token. The code itself is obtained from the authorization server where the user gets a chance to see what information the client is requesting, and approve or deny the request.
Open the following URL in your browser:
https://aai-demo.egi.eu/oidc-client
and click on “Authorise”.

Authorization Code Flow
Authorization Code Flow

Choose which Identity Provider you wish to authenticate and enter your credentials.
Authorization Code Flow

Give your consent to your IdP to pass your information to EGI Check-in.
If it is your first time you are using EGI Check-in, you will need to register an account. Click on “BEGIN” to start the sign up flow.
Review your information, agree the Terms of Use and click on “Submit”.
Next, you will need to confirm your email address. Check your inbox and open the link on your browser.
Click on “Accept” to finalize the registration.
Authorization Code Flow

Give your consent to EGI Check-in to pass your information to the DEMO Client.
Authorization Code Flow

The DEMO Client will display your Access Token and your Refresh Token.
Obtaining information from UserInfo endpoint

To get the user information of this token, copy the value of the highlighted text box and paste if to your terminal.

Client ID: f68f5df4-d1b1-44b1-a93d-4aab8f80e5c4
Client Secret: AMRnxknUXwJ3nNGC2Xz2NuLt-pBJr3eJRBF9jKORuJ2Zmzb7Nzqwh4k5yRYRomeRzx5Y847twEDUNfWxxwQSUg

Access Token:
eyJrWQOiJvaWRjIiwYXnlojUIMyNTYfQ.ejzdWIOjIjZAwOWW4N2NlMRmMGE2OWZ

To get the user info use the following curl command:
curl -H 'Authorization: Bearer eyJrWQOiJvaWRjIiwYXnlojUIMyNTYfQ.ejzdWIOjIjZAw
To introspect the token use the following curl command:
curl -u 'f68f5df4-d1b1-44b1-a93d-4aab8f80e5c4' AMRnxknUXwJ3nNGC2Xz2NuLt-pBJr
NOTE: New access tokens expire in 1 hour.

Refresh Token:
eyJhbGciOiJub25lIn0.ejylHEAhIqIEOdeK1Nzc1MzgsImpOaSI6jUxN2ZkJyJlWm3NjtHNDRINS
NOTE: New refresh tokens expire in 13 months.

To generate access tokens from this refresh token use the following curl command:
curl -X POST -u 'f68f5df4-d1b1-44b1-a93d-4aab8f80e5c4' AMRnxknUXwJ3nNGC2Xz2NuLt-pBJr
NOTE: New access tokens expire in 1 hour.

You can manage your refresh tokens in the following link:
https://aai-demo.egi.eu/oidc/manage/user/services
Obtaining information from UserInfo endpoint

```
grnet@GRNETs-MacBook-Pro-2 $ curl -H 'Authorization: Bearer eyJraWQOJcW3vNR1L5wYXk6Jj0UMyNtYlFQ.eyJzd10LYLjJw0W4M4N2 NUNDRmWGEZ2M3XwnWjOg3nxdUDk13NDAW9xkvWvWf7ildne8LiZjMKjZWY3ncz3HtvLjNvQq6VvA5ZldSi1mF6vCo2vQfZQWEjYr4vdRlNlV1Q3Tnk LTRhYW1z7Jw9ZTVcISinLzcyi6ln6fB8f0LwwX95hWktzQ1by8L2kxZxvC1Lz9PZG_NClytS1VACi6MTU40TU1Mj3z0Cm 내용FBJvoxNTOQZMN2aLCjxGG kU0L3jztNvmSrR0RHyT6MwL4T1YmQq6YnnhnmnoD12JzlvfuUKy4wMrVr1ayAdu2229o2fjxS7qkac101EF2X1af_LKFW-V_cjLJLM6zhm6k52134pdxnD Juby10nyj13cCjkgvwpdTHq2gCj1hldx7pml5pOnhj9q1r00d296-9Y///z0l10uyQVH93O8rKwKCIimf3o6-264q4Ywclm_4Vf/g52qDQz.B8BN-KS4EF03YrPeDKN2j_UkdvOF64zu12xXQ01l-1AqQTs0SbnB5nq7/a6Lx1k3YQ9CdJEFYFZnI-1LCLm5Sgde4sb0b0CUBFb11ZjloSHDR1Bhe3xeaQAd7yX1BSTYK6S3C0xcFbE_knpVKB bnp7f4E8quyiy7v7wJQ" -H 'Content-type: application/json' https://aal-demo.egi.eu/oicd/userinfo | python -m json.tool
```

```
{
   "acr": "https://aal.eu.org/LoA#Substantial",
   "eduperson_assurance": ["https://aal.eu.org/LoA#Substantial"
   ],
   "eduperson_entitlement": [
      "urn:naace:egi.eu:group:registry:test-group:role:member#aal.eu",
      "urn:naace:egi.eu:group:demofedcloud.eu:members:role:member#aal.eu",
   ],
   "edupersonScoped_affiliation": [
      "member@grnet-hq.admin.grnet.gr"
   ],
   "email": "nikosiev@admin.grnet.gr",
   "family_name": "Evangelou",
   "given_name": "Nikolaos",
   "name": "Nikolaos Evangelou",
   "preferred_username": "nevangelou",
   "sub": "bf809c87cb04f0a69f2b3c98767147eb3b748bbedaeef07b78ef33ef777318e610@egi.eu"
}
```
Introspecting tokens

To introspect the token, copy the value of the highlighted text box and paste if to your terminal.
Refreshing token

To create an access token using a refresh token, copy the value of the highlighted text box and paste if to your terminal.
Refreshing token

```python
grnet@GRNETs-MacBook-Pro-2:~$ curl -X POST -u 'f68f5df4-d1b1-44b1-a93d-4a4b8f80e5c4':'AMRxnxUxnJ3n9NC2xzZ2NuLt-pBJr3eJRBFD9fK40H2Mzb7N7zqwh4k58yBOMeRx58yS87iWDeO UntFxxwQSUGG' -d 'client_id=f68f5df4-d1b1-44b1-a93d-4a4b8f80e5c4&client_secret=AMRxnxUnKwx3Jn9NC2x2zZ2NuLt-pBJr3eJRBFD9fK40H2Mzb7N7zqwh4k58yBOMeRx58yS87iWDeO UntFxxwQSUGG&grant_type=refresh_token&refresh_token=eyJhbGciOiJub25lIn0.eyJleHAiOiJ0dk10DA1MzIsIm5p6aSI61JJa2NGQ0DKi1LWQ5DMtNDN4I5S05OTQ2LW3MNYYwNTMzNjdyYyJ9.&scope=openid%20email%20profile' https://aai-demo.egi.eu/oidc/token | python -n json.tool;
```
Live Demo

Device Code Flow
Device Code Flow

The Device Code grant type is used by browserless or input-constrained devices in the device flow to exchange a previously obtained device code for an access token.
Device Code Flow

Type the following in your terminal:

curl -H 'Content-Type: application/x-www-form-urlencoded' -X POST 'https://aai-demo.egi.eu/oidc/devicecode' -d 'client_id=f68f5df4-d1b1-44b1-a93d-4aab8f80e5c4&scope=openid%20email%20profile' | python -m json.tool;
Open the “verification_uri” in your browser and enter the “user_code”
Then approve the request.
Back to the terminal, create a variable for the “device_code”.

```bash
export device_code=46ff7852-6c39-444a-8fc7-86e46e85d34c
```
Type the following in your terminal:

curl -H 'Content-Type: application/x-www-form-urlencoded' -X POST 'https://aai-demo.egi.eu/oidc/token' -d 'grant_type=urn:ietf:params:oauth:grant-type:device_code' -d "device_code=${device_code}" -d 'client_id=f68f5df4-d1b1-44b1-a93d-4aab8f80e5c4' -d 'client_secret=AMRnkxnUWxJ3nN9C2XzZ2NuLt-pBJRr3eJRBF9jfK0rUH2Zmb7Nzqwh4k58YRoMeRx5Y847IwEOUNtFWxwQSUGg' -d 'scope=openid email profile' | python -m json.tool;
Device Code Flow
Live Demo

Proxy X.509 certificate retrieval using SSH key authentication
Proxy X.509 certificate retrieval using SSH key authentication

- EGI MasterPortal also allows users to authenticate using their SSH key pair to retrieve proxy certificates issued by the RCauth Online CA
Proxy X.509 certificate retrieval using SSH key authentication

- Users need to first upload the public key via a self-service portal, https://aai.egi.eu/sshkeys/
Proxy X.509 certificate retrieval using SSH key authentication

- Users need to first upload the public key via a self-service portal, https://aai.egi.eu/sshkeys/
Proxy X.509 certificate retrieval using SSH key authentication

- Users can then obtain a proxy certificate by doing:

  `ssh proxy@ssh.aai.egi.eu`
Proxy X.509 certificate retrieval using SSH key authentication

• About once a week users need to follow a web-flow to ensure a long-lived proxy certificate is present in MasterPortal

• Demo VO Portal: https://aai.egi.eu/vo-portal/
Proxy X.509 certificate retrieval using SSH key authentication

- Users can then obtain a proxy certificate by doing:

```bash
ssh proxy@ssh.aai.egi.eu
```
Check-in

Coming soon

• Improved end-user experience
  ▪ New user enrolment UI
  ▪ Identity linking
  ▪ IdP discovery (hinting)

• Improved integration for service providers
  ▪ Service accounts
  ▪ Extended OAuth2 token introspection validation in multi-domain scenarios
  ▪ New service registration web portal for SAML & OIDC services
Check-in Documentation

• **Usage guide**

• **Integration guide for service providers**

• **Integration guide for identity providers**

• **Frequently Asked Questions**
Thank you for your attention.

Questions?