

Using EOSC for on demand computational service

Stéphane Aicardi

IVOA, Bologna, May 2023





EOSC is an initiative founded by the European commission.

Provides :

- portal to find resources (datasets, publications, software...)
- computing and storage resources

Here we use the cloud computing service handled by **EGI** federated cloud.

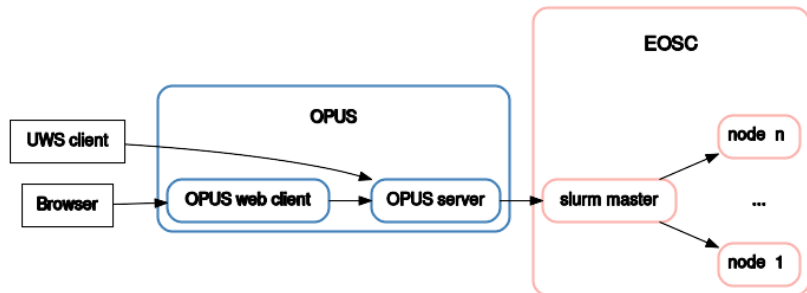
Goals of this presentation

Use the computing power offered by EOSC to provide on demand computation service.

Make a responsible use of the cloud: don't waste common resources when we don't need them.

This work has been done for the VESPA Cloud task in Europlanet 2024.

How does it work?



OPUS is a UWS job manager with a web client.

Slurm is a job scheduling system devised for HPC clusters.

The number of nodes limits the admissible load.

Most of the deployment of Opus, slurm master and computational nodes is done by scripts [here](#).

Note: for a large cluster, it would be better to create an openstack image.

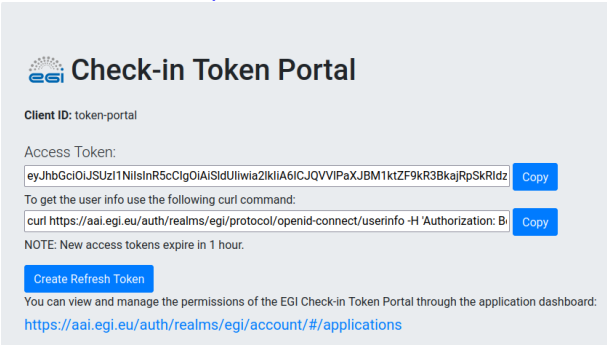
Some of the technical details

Extract from `slurm.conf` :

```
SuspendProgram=/usr/local/sbin/SlurmSuspendProgram
ResumeProgram=/usr/local/sbin/SlurmResumeProgram
SuspendExcNodes=node[2-NB_NODES]
SuspendTimeout=30
ResumeTimeout=600
SuspendTime=3600
```

Those two programs setup authentication w.r.t EOSC and stop or starts predeployed nodes.

- authenticate on EGI portal

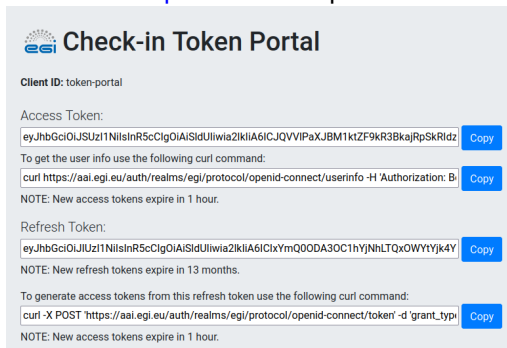


The screenshot shows the 'Check-in Token Portal' interface. At the top left is the EGI logo. The main heading is 'Check-in Token Portal'. Below this, it displays 'Client ID: token-portal'. There is a section for 'Access Token:' with a text input field containing a long alphanumeric string and a 'Copy' button to its right. Below the token field, it says 'To get the user info use the following curl command:' followed by a text input field containing a curl command and another 'Copy' button. A note states 'NOTE: New access tokens expire in 1 hour.' Below the note is a blue button labeled 'Create Refresh Token'. At the bottom, there is a message: 'You can view and manage the permissions of the EGI Check-in Token Portal through the application dashboard:' followed by a blue hyperlink: <https://aai.egi.eu/auth/realms/egi/account/#/applications>.

- use the access token in fedcloud to setup openstack environment
 - openstack allows creating/starting/stopping/destroying VMs
- access token valid for 1h => not suitable for automatic service

Slurm/cloud interaction

- authenticate on [EGI portal](#) and request a refresh token



egi Check-in Token Portal

Client ID: token-portal

Access Token:

```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUiIiwia2kiOiA6IiCjQVVlPaXJBM1ktZF9kR3BkaWpScSkRldz
```

[Copy](#)

To get the user info use the following curl command:

```
curl https://aai.egi.eu/auth/realms/egi/protocol/openid-connect/userinfo -H 'Authorization: B
```

[Copy](#)

NOTE: New access tokens expire in 1 hour.

Refresh Token:

```
eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUiIiwia2kiOiA6IiCixYmQ0DA3OC1hYjNhLTQxOWYyYjk4Y
```

[Copy](#)

NOTE: New refresh tokens expire in 13 months.

To generate access tokens from this refresh token use the following curl command:

```
curl -X POST https://aai.egi.eu/auth/realms/egi/protocol/openid-connect/token -d 'grant_type
```

[Copy](#)

NOTE: New access tokens expire in 1 hour.

- the refresh token is valid for 13 months!
- store it in a specific location on the server
- scripts can use the refresh token to generate an access token when needed (e.g. when slurm wants to start or stop a node)