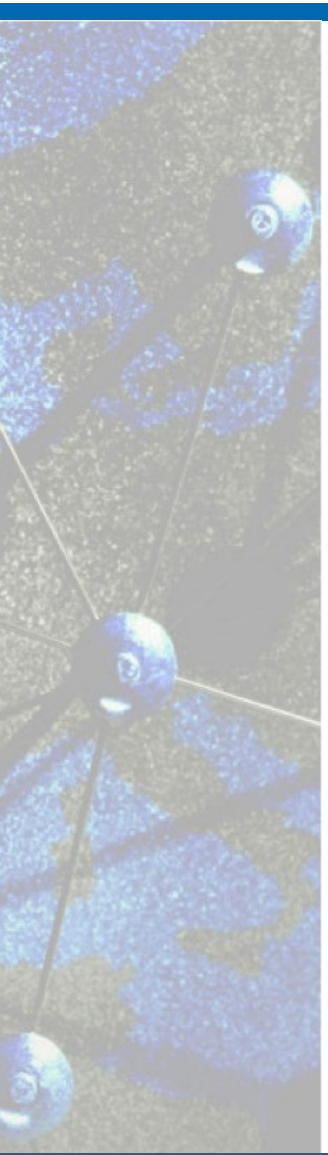


Federated AAI with VOMS and Token Profile Extending HTTP Authentication

Björn Hagemeier

Keystone

- OpenStack service for
  - Identity, Token, Catalog and Policy
- Intended to be used as an interface in front of existing backend services
  - LDAP etc.
  - provides own user management, too
- Supports multiple mechanisms
  - username/password
  - (VOMS proxy) certificates

HTTP authentication

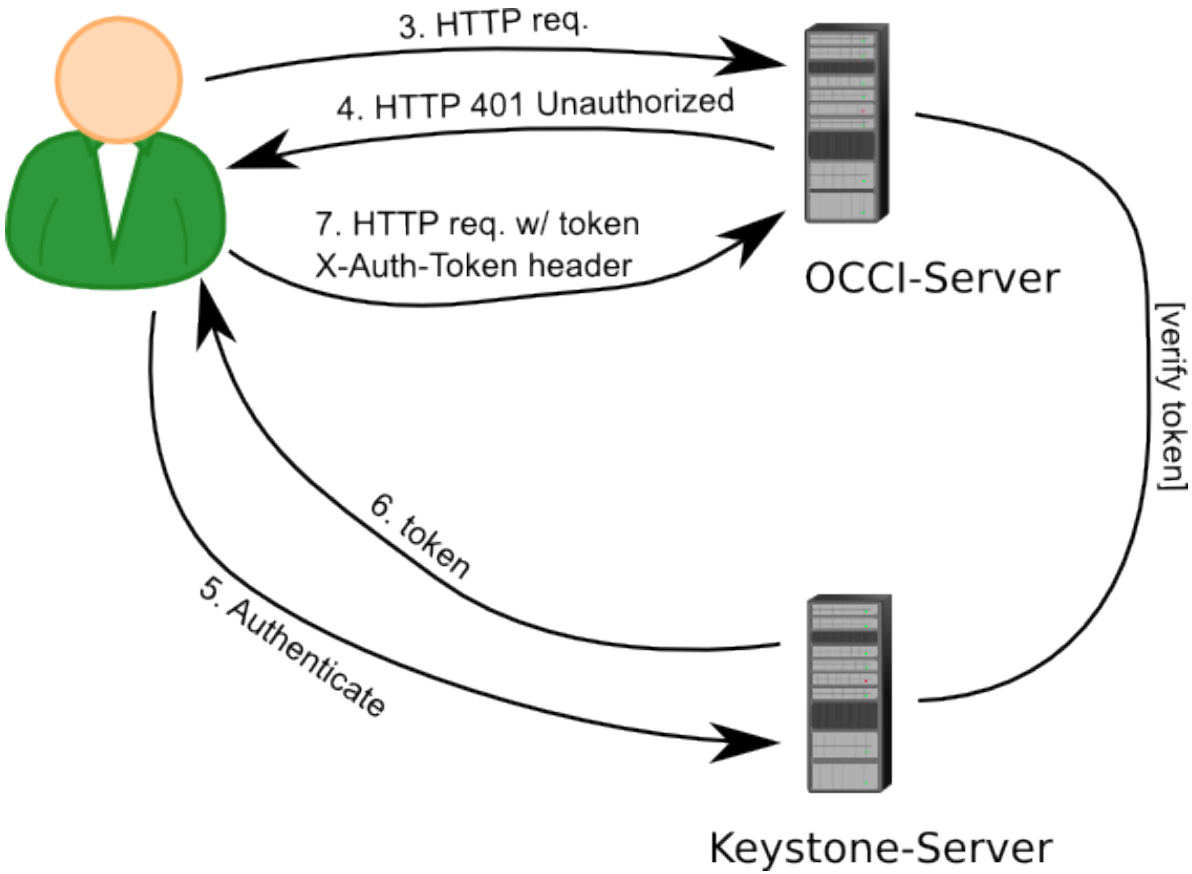
- RFC 2616 (HTTP 1.1) defines “www-authenticate” header
- RFC 2617 defines Basic and Digest authentication schemes
- Does not preclude other schemes
  - Keystone makes use of this
  - Refers client to respective authentication service for this service
  - i. e. `'WWW-Authenticate: Keystone uri=''https://keystone....'' '`

HTTP Protocol

```
> GET /-/ HTTP/1.1
> User-Agent: curl/7.26.0
> Host: egi-cloud.zam.kfa-juelich.de:8787
> Accept: */*
>

< HTTP/1.1 401 Unauthorized
< Date: Fri, 13 Sep 2013 12:04:20 GMT
< WWW-Authenticate: Keystone uri='https://egi-cloud.zam.kfa-juelich.de:5000'
< Content-Length: 381
< Content-Type: text/html; charset=UTF-8
< Vary: Accept-Encoding
<
...
```

Interaction



Benefit:

- Services only need to know Keystone AuthN mechanism

Keystone API

- POST v2.0/tokens – Authenticates and generates a token.
- GET/HEAD v2.0/tokens/{tokenId}?belongsTo=string – Validate token (belonging to specific tenant)

Payload

```
{
   "auth":{
      "passwordCredentials":{
         "username":"test_user",
         "password":"mypass"
      },
      "tenantName":"customer-x"
   }
}
```

- Tenant is a “unit of ownership” in Keystone

HTTP to Service

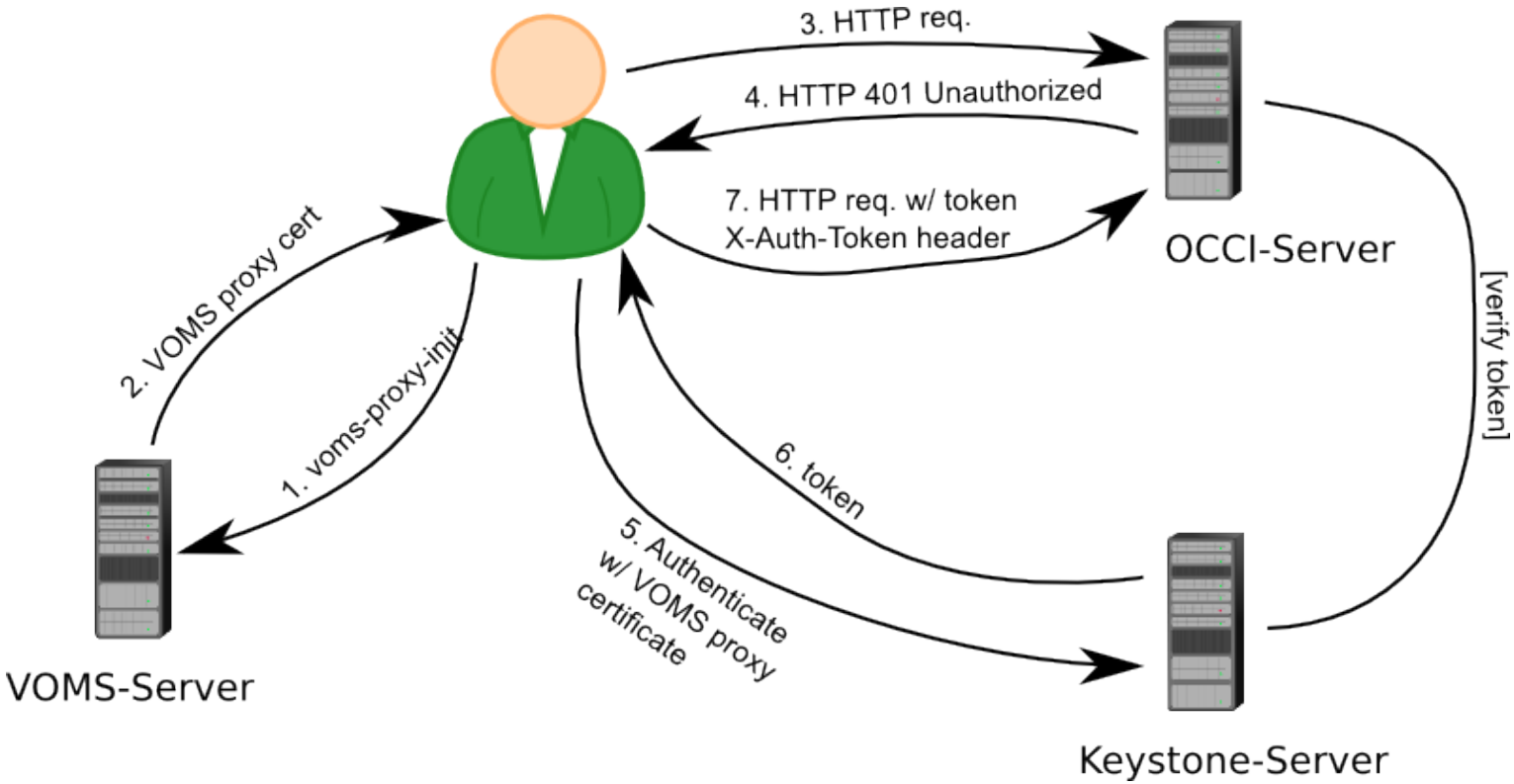
```
> GET /-/ HTTP/1.1
> User-Agent: curl/7.26.0
> Host: egi-cloud.zam.kfa-juelich.de:8787
> Accept: */*
> X-Auth-Token: 4fb2ca8c569547e8b88f6d1cfeac6547
>

< HTTP/1.1 200 OK
< Date: Fri, 13 Sep 2013 12:38:40 GMT
< Server: pyssf OCCI/1.1
< Content-Length: 18159
< Content-Type: text/plain
< Vary: Accept-Encoding
<
...
```

FedCloud integration

- Authenticate using VOMS proxy certificate

Interaction



VOMS Enabled Keystone API

- Make call using SSL client authentication using VOMS proxy certificate
- Map VOs to tenants
- Enabling a VO means creating a mapping for it

Payload

```
{
   "auth":{
      "voms": true
   }
}
```

- VO membership extracted from VOMS attributes

Thank you

- More information
  - Federated AAI Configuration page: `https://wiki.egi.eu/wiki/Federated_AAI_Configuration#OpenStack`
  - Generic documentation about VOMS integration: `http://keystone-voms.readthedocs.org/en/latest/`
- Kudos
  - Álvaro López García