

Deploying federated services in eduroam with Moonshot

Grupos de trabajo de RedIRIS 2014, Madrid

Alejandro Pérez, Rafael Marín, Gabriel López

Departamento de Ingeniería de la Información y las Comunicaciones
University of Murcia

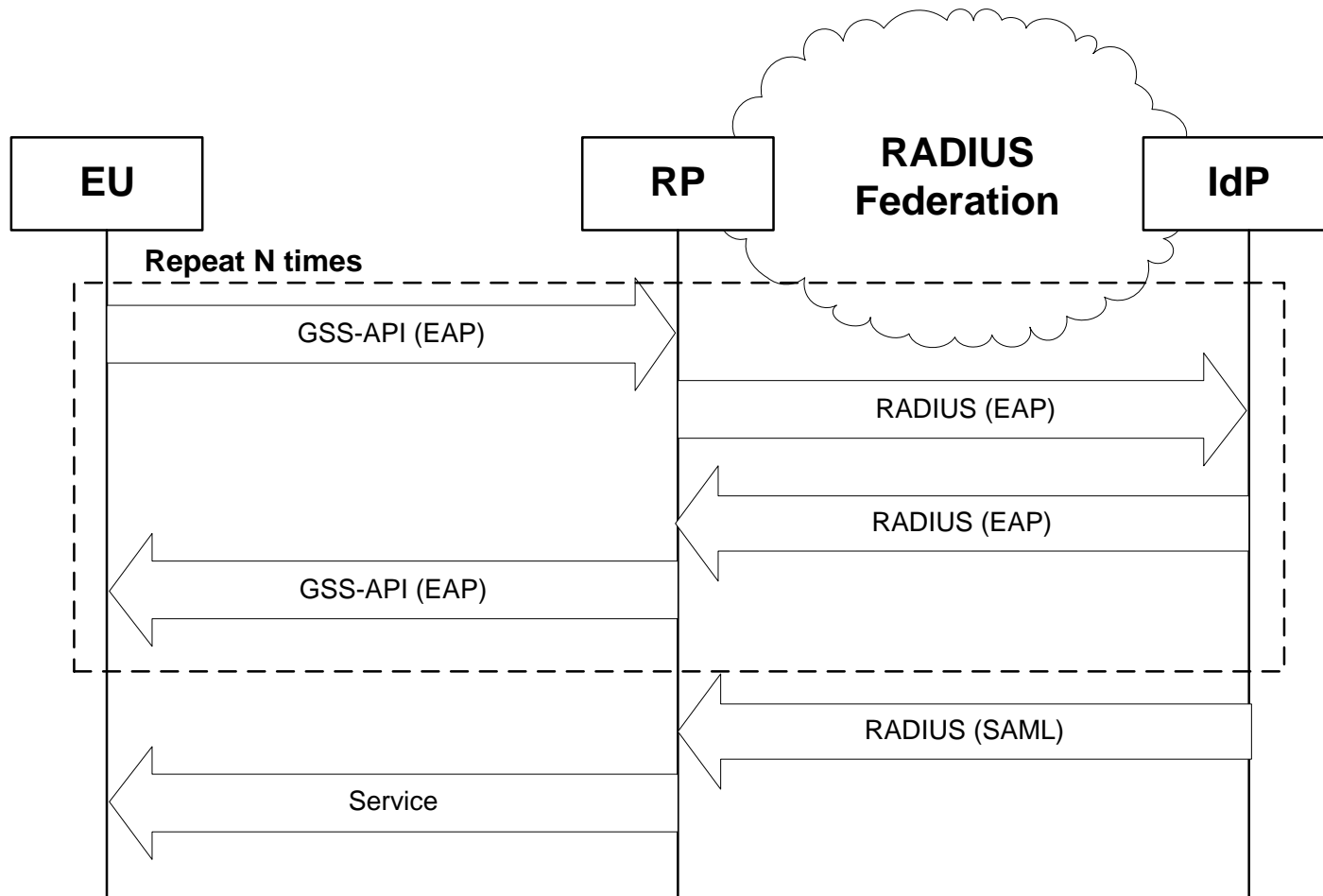
Motivation

- Identity federations
 - Trust relationships to identify end users
 - Usability and lower deployment costs
- Drawbacks
 - Defined for specific kinds of services
 - Use of different technologies
 - Access to the network service (e.g. eduroam) → RADIUS, Diameter...
 - Web applications → SAML, OpenID, OAuth...
 - Some services do not provide support for federation
 - Email, remote file access, remote terminal access...

What's Moonshot?

- Moonshot
 - Development of a technology to bring the identity federation concept to any kind of service (e.g. cloud, ftp, http, ssh...)
- Key components:
 - EU → Wants to access a service
 - RP → Provides the service
 - IdP → Authenticate the end user and provides authorization information to the RP
- Key technologies:
 - GSS-API → Access control to the service (between EU and RP)
 - RADIUS → Federation (between RP and IdP)
 - SAML → Authorization (between RP and IdP)
 - EAP → User authentication (between EU and IdP)

What's Moonshot?



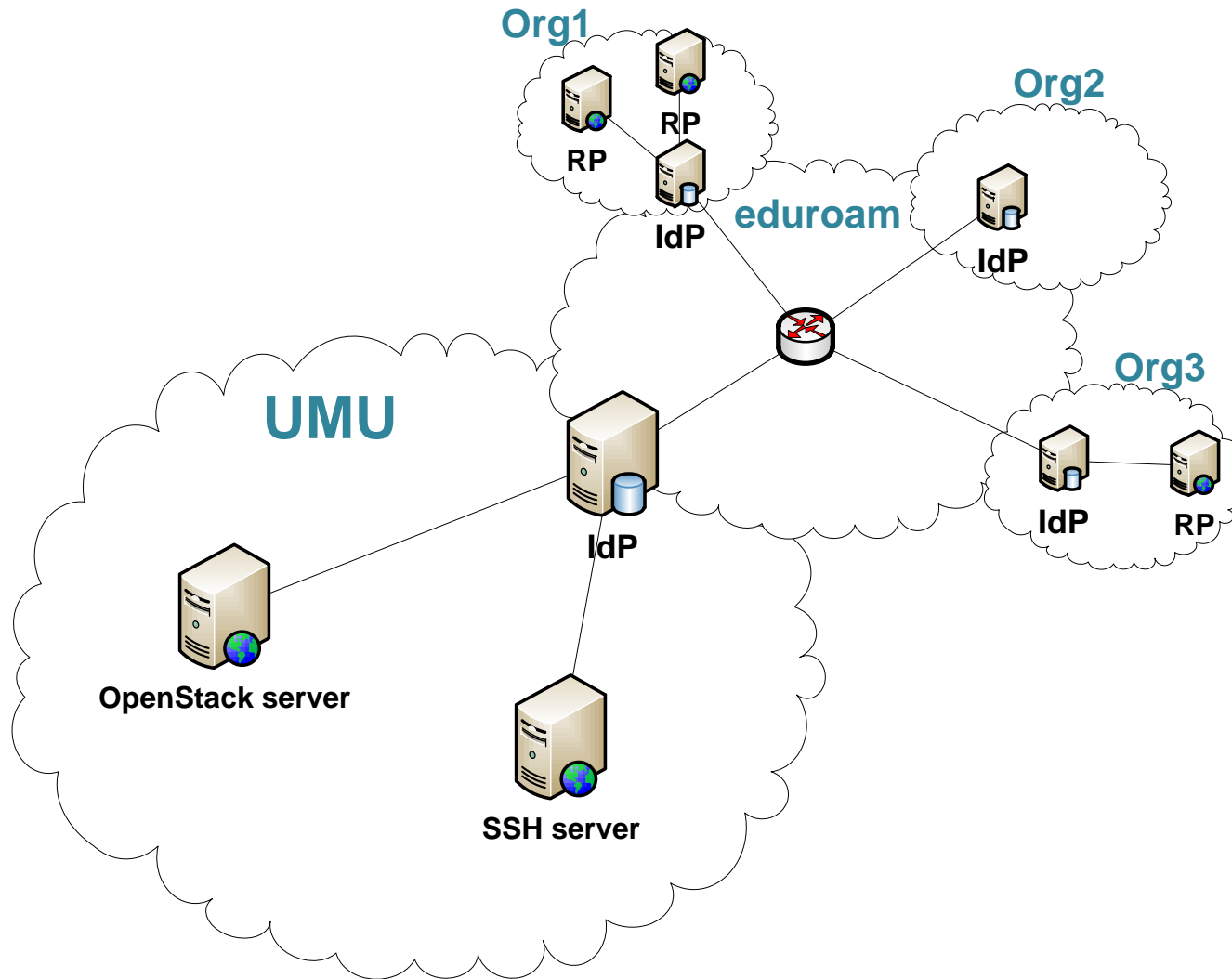
What's Moonshot?

- Partly performed within GN3 project
 - Participated by RedIRIS and UMU
- Being standardized within the IETF (ABFAB WG)
- Completely implemented, documented, and maintained by the Moonshot community
 - <https://community.ja.net/groups/moonshot>

Deploying Moonshot

- The RADIUS infrastructure of eduroam is an ideal candidate to deploy Moonshot
 - Trust relationships are already established
 - A great number of interconnected organizations
- We'll see how to deploy the different components of Moonshot using this infrastructure
- Two practical examples, deployed at UMU:
 - SSH server allowing the access to the *federated* account to any member of the eduroam community (**GN3Plus**)
 - OpenStack server allowing the access to the *swiftenanttest1* tenant only to UMU's members, and the access to the *swiftenanttest2* tenant to any member of the eduroam community (**CLASSe**)

Deploying Moonshot



Deploying Moonshot - IdP

- Any current RADIUS server from the eduroam network can act as a Moonshot IdP...
 - But they will not send the SAML assertion
- To configure a new IdP:
 1. Install FreeRADIUS and connect it to the eduroam's infrastructure
 2. Create the required user accounts
 3. Configure FreeRADIUS to generate a SAML assertion:
 - Static → Fixed assertion template, filled with FreeRADIUS variables
 - Dynamic → Assertion generated with OpenSAML, filled with values obtained from different data bases (in development)

Examples #1 and #2: IdP

- Configure a new RADIUS server
 - UM's subdomain
 - moonshot.um.es
- Create a testing account
 - test@moonshot.um.es
- Configure the SAML assertion template
 - post-auth section of sites-enabled/default file

```
update reply {  
  SAML-AAA-Assertion = "<saml:Assertion xmlns:saml='urn:oasis:names:tc:.....'"  
  SAML-AAA-Assertion += "<saml:Conditions NotOnOrAfter='2015-03-19T08:30:00Z'/>"  
  SAML-AAA-Assertion += "<saml:Issuer>moonshot.inf.um.es</saml:Issuer>"  
  .....
```

Examples #1 and #2: IdP

```
<saml:Assertion xmlns:saml='urn:oasis:names:tc:SAML:2.0:assertion' ...>
  <saml:Conditions NotOnOrAfter='2015-03-19T08:30:00Z' />
  <saml:Issuer>moonshot.um.es</saml:Issuer>
  <saml:Subject>
    <saml:NameID Format='urn:oasis:names:tc:SAML:2.0:nameid-format:transient'>
      %{%reply:User-Name}:-{%request:User-Name}}
    </saml:NameID>
  </saml:Subject>
  <saml:AttributeStatement>
    <saml:Attribute Name='studentcard' ...>
      <saml:AttributeValue>Student</saml:AttributeValue>
    </saml:Attribute>
    <saml:Attribute Name='affiliation' ...>
      <saml:AttributeValue>umu</saml:AttributeValue>
    </saml:Attribute>
  </saml:AttributeStatement>
</saml:Assertion>
```

Example:
test@moonshot.um.es

Deploying Moonshot - RP

- Any application supporting the GSS-API should work with Moonshot
 - Although some of them are badly programmed and require small adjustments (e.g. OpenSSH)
- It will be needed to:
 1. Install Moonshot's code in the RP
 2. Configure a RADIUS proxy connected to the eduroam's infrastructure
 3. Configure attribute mapping (authorization)
 - Convert RADIUS and/or SAML attributes into the application's specific attributes

Example #1: SSH server

- Configure the machine:
 - moonshot-ssh.inf.um.es
- Install Moonshot
- Install OpenSSH server patched for Moonshot
 - Available from the Moonshot repositories
- Configure the RADIUS proxy
 - moonshot.um.es
- Configure attribute mapping
 - If TRUE →
 - OpenSSH.local_login_user := federated
 - Does not require SAML assertion
 - This authorizes any user to access to the account federated@moonshot-ssh.inf.um.es

Example #2: OpenStack server

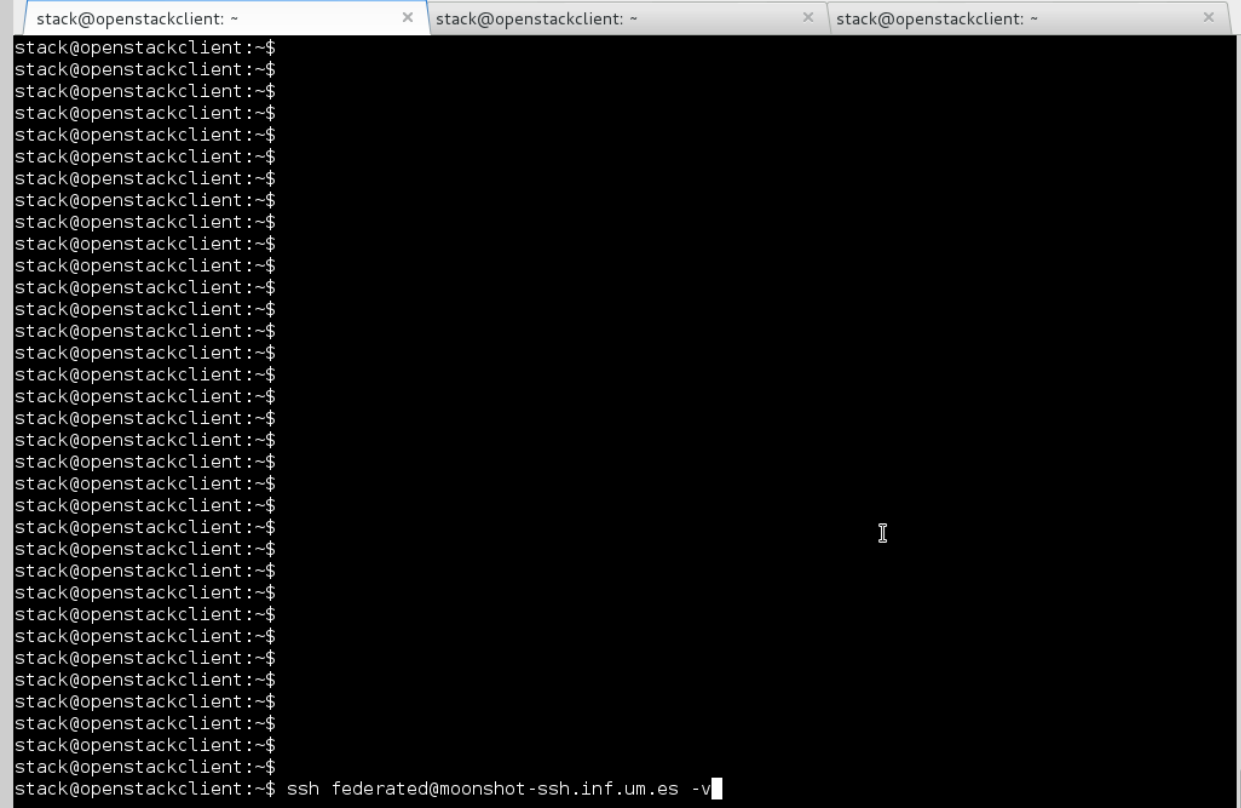
- Configure the machine
 - classe1.qalab.geant.net
- Install Moonshot
- Install OpenStack server with support for GSS-API
 - <https://github.com/kwss/keystone>
- Configure the RADIUS proxy
 - moonshot.um.es
- Configure attribute mapping
 - If *SAML.affiliation* == umu →
 - OpenStack.tenant := swifttenanttest1
 - Else →
 - OpenStack.tenant := swifttenanttest2

Deploying Moonshot - EU

- Any application supporting GSS-API should work with Moonshot
- It will be needed to:
 1. Install Moonshot code
 2. Try to access the service
 3. Introduce or select the desired identity from the selector

Example #1: SSH client

1. Install Moonshot
2. Access to the service
 - `ssh federated@moonshot-ssh.inf.um.es`

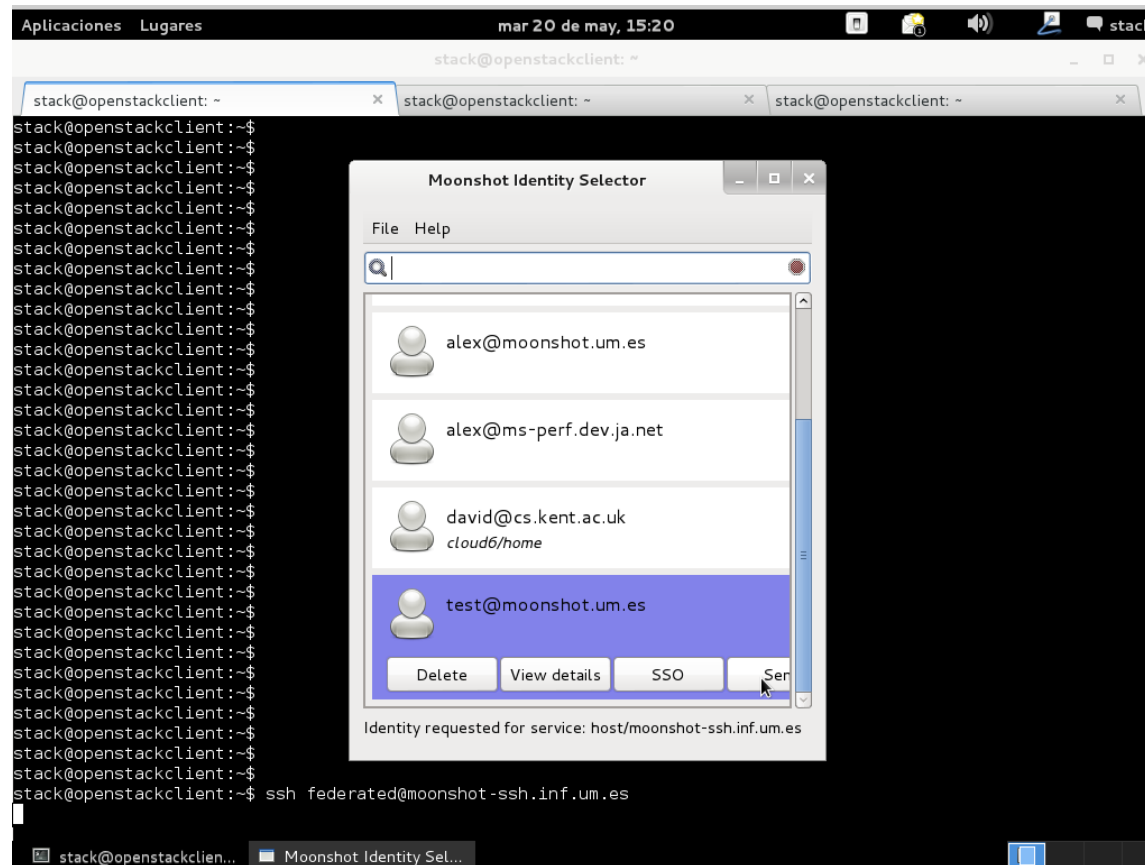


```
stack@openstackclient: ~  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$  
stack@openstackclient: ~$ ssh federated@moonshot-ssh.inf.um.es -v
```

Example #1: SSH client

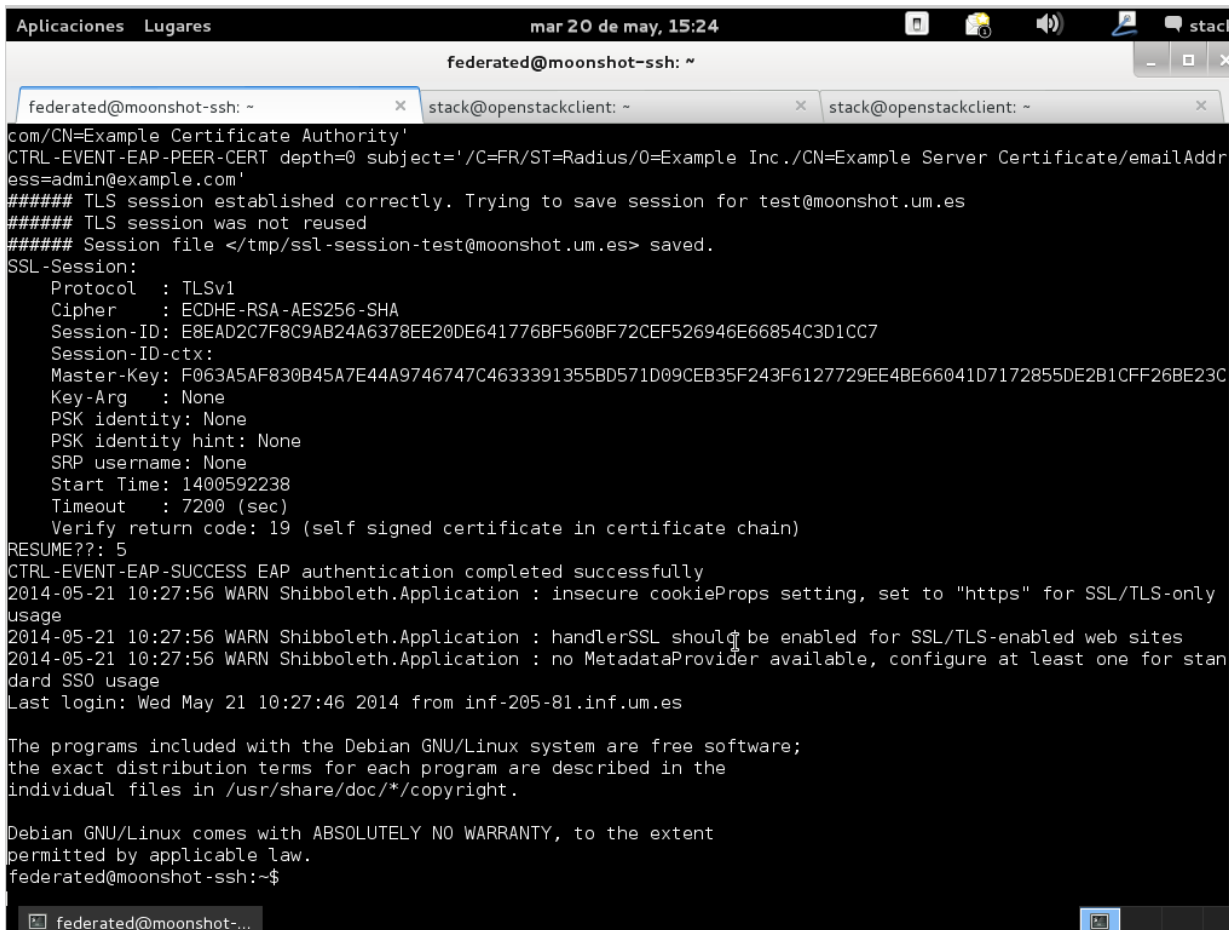
3. Select the identity

- test@moonshot.um.es
- Or any other identity valid within eduroam



Example #1: SSH client

4. Access to the requested service



```
mar 20 de may, 15:24
federated@moonshot-ssh: ~
federated@moonshot-ssh: ~
stack@openstackclient: ~
stack@openstackclient: ~
com/CN=Example Certificate Authority'
CTRL-EVENT-EAP-PEER-CERT depth=0 subject='/C=FR/ST=Radius/O=Example Inc./CN=Example Server Certificate/emailAddress=admin@example.com'
##### TLS session established correctly. Trying to save session for test@moonshot.um.es
##### TLS session was not reused
##### Session file </tmp/ssl-session-test@moonshot.um.es> saved.
SSL-Session:
  Protocol : TLSv1
  Cipher   : ECDHE-RSA-AES256-SHA
  Session-ID: E8EAD2C7F8C9AB24A6378EE20DE641776BF560BF72CEF526946E66854C3D1CC7
  Session-ID-ctx:
  Master-Key: F063A5AF830B45A7E44A9746747C4633391355BD571D09CEB35F243F6127729EE4BE66041D7172855DE2B1CFF26BE23C
  Key-Arg   : None
  PSK identity: None
  PSK identity hint: None
  SRP username: None
  Start Time: 1400592238
  Timeout   : 7200 (sec)
  Verify return code: 19 (self signed certificate in certificate chain)
RESUME??: 5
CTRL-EVENT-EAP-SUCCESS EAP authentication completed successfully
2014-05-21 10:27:56 WARN Shibboleth.Application : insecure cookieProps setting, set to "https" for SSL/TLS-only usage
2014-05-21 10:27:56 WARN Shibboleth.Application : handlerSSL should be enabled for SSL/TLS-enabled web sites
2014-05-21 10:27:56 WARN Shibboleth.Application : no MetadataProvider available, configure at least one for standard SSO usage
Last login: Wed May 21 10:27:46 2014 from inf-205-81.inf.um.es

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
federated@moonshot-ssh:~$
```

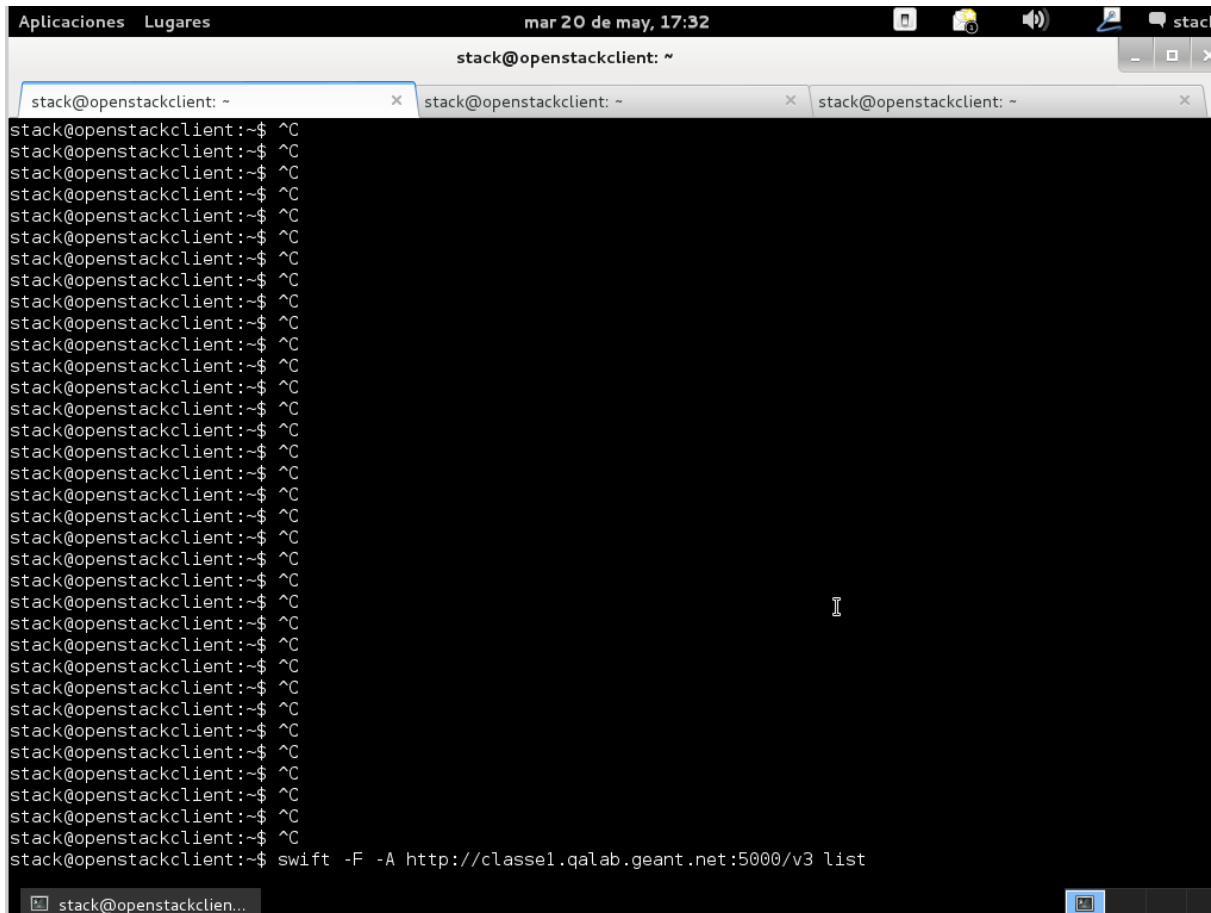
Example #2: OpenStack client

1. Install Moonshot
2. Install the OpenStack client with support for the GSS-API
 - <http://sec.cs.kent.ac.uk/demos/keystone.html>

Example #2: OpenStack client

3. Access to the service

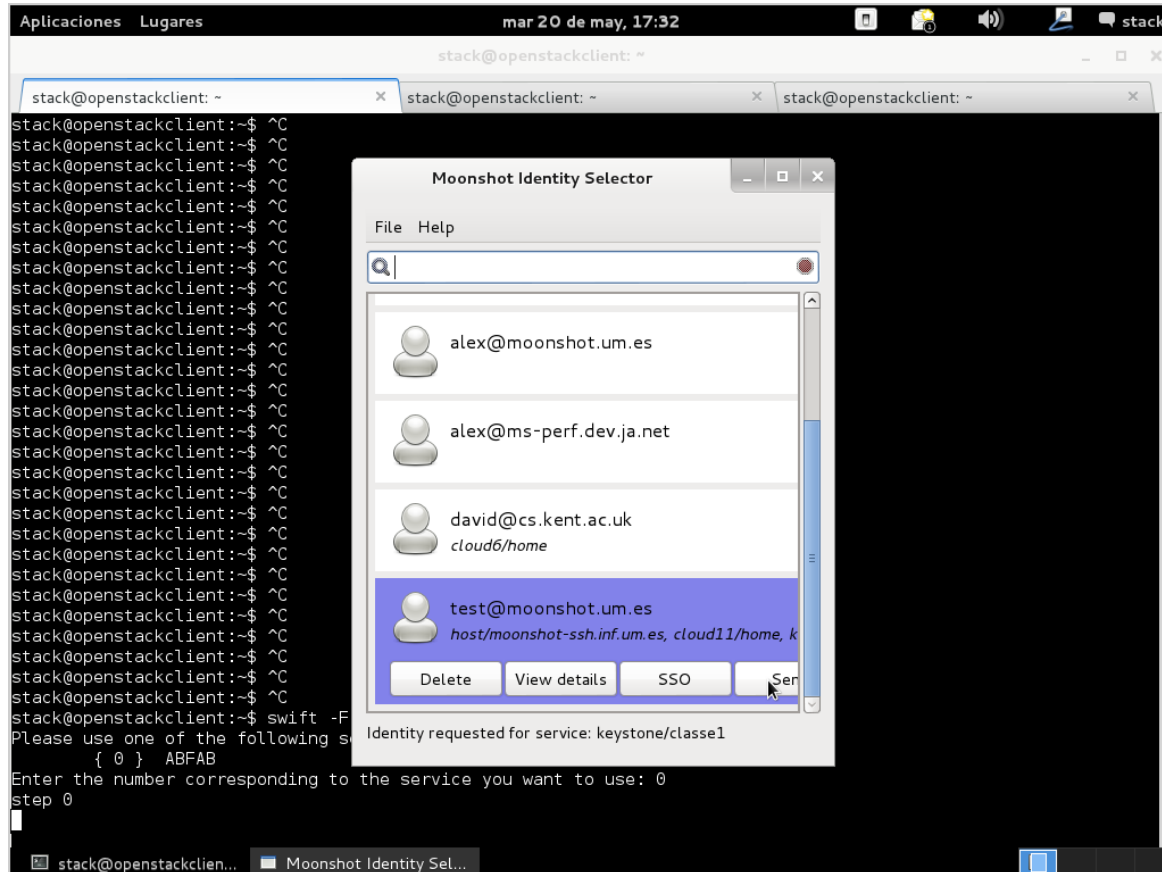
- `swift -F -A http://classe1.qalab.geant.net:5000/v3 list`



The screenshot shows a terminal window titled "stack@openstackclient: ~" with a system clock of "mar 20 de may, 17:32". The terminal displays a series of shell prompts "stack@openstackclient:~\$" followed by a cursor. At the bottom, the command `swift -F -A http://classe1.qalab.geant.net:5000/v3 list` is entered. The terminal window is part of a desktop environment with a taskbar at the bottom and a system tray at the top right.

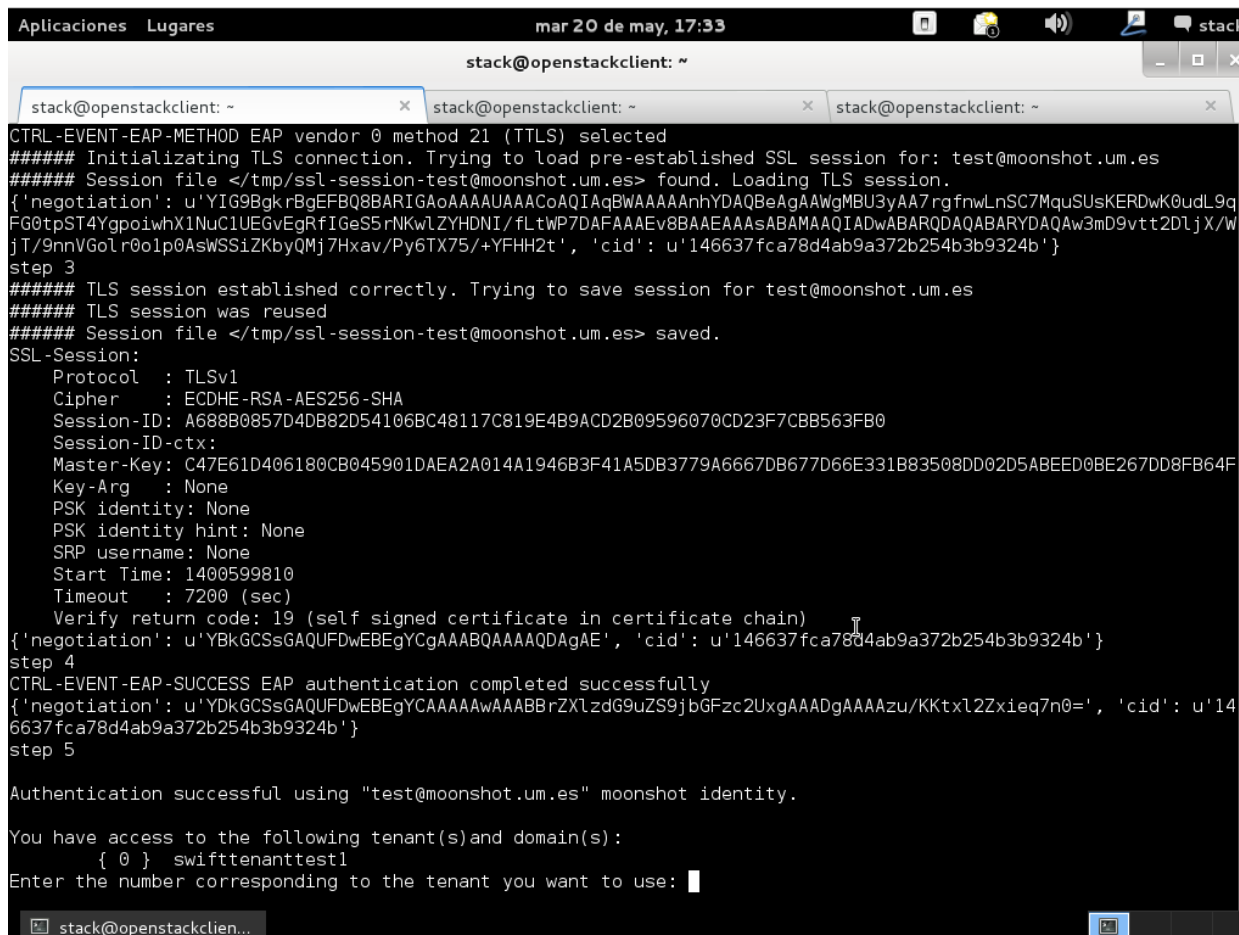
Example #2: OpenStack client

5. Select the desired identity
 - test@moonshot.um.es
 - Or any other identity valid within eduroam



Ejemplo 2: Cliente OpenStack

6. Get access to the *tenant*



```
Aplicaciones Lugares mar 20 de may, 17:33 stack
stack@openstackclient: ~
stack@openstackclient: ~
CTRL-EVENT-EAP-METHOD EAP vendor 0 method 21 (TTLS) selected
##### Initializing TLS connection. Trying to load pre-established SSL session for: test@moonshot.um.es
##### Session file </tmp/ssl-session-test@moonshot.um.es> found. Loading TLS session.
{'negotiation': u'YIG9BgrBgEFBQ8BARIGAoAAAAUAACoAQIAqBwAAAAAnhYDAQBeAgAAWgMBU3yAA7rgfnwLnSC7MquSUsKERDwK0udL9q
FG0tpST4YgpoiwhX1NuC1UEGvEgRfIGeS5rNKwLZYHDNI/fLtWP7DAFAAAEv8BAEEAAsABAMAAQIADwABARQDAQABARYDAQAw3mD9vtt2DljX/W
jT/9nnVGolr0o1p0AsWSSiZKbyQMj7Hxav/Py6TX75/+YFH2t', 'cid': u'146637fca78d4ab9a372b254b3b9324b'}
step 3
##### TLS session established correctly. Trying to save session for test@moonshot.um.es
##### TLS session was reused
##### Session file </tmp/ssl-session-test@moonshot.um.es> saved.
SSL-Session:
  Protocol      : TLSv1
  Cipher       : ECDHE-RSA-AES256-SHA
  Session-ID  : A688B0857D4DB82D54106BC48117C819E4B9ACD2B09596070CD23F7CBB563FB0
  Session-ID-ctx:
  Master-Key  : C47E61D406180CB045901DAEA2A014A1946B3F41A5DB3779A6667DB677D66E331B83508DD02D5ABEE00BE267DD8FB64F
  Key-Arg     : None
  PSK identity: None
  PSK identity hint: None
  SRP username: None
  Start Time  : 1400599810
  Timeout     : 7200 (sec)
  Verify return code: 19 (self signed certificate in certificate chain)
{'negotiation': u'YBkGCSsGAQUFDwEBEGYCAAAABQAAAAQDAgAE', 'cid': u'146637fca78d4ab9a372b254b3b9324b'}
step 4
CTRL-EVENT-EAP-SUCCESS EAP authentication completed successfully
{'negotiation': u'YDkGCSsGAQUFDwEBEGYCAAAAwAAABBrZXlzdG9uZS9jbGZzc2UxgAAADgAAAAzu/KKtxL2Zxleq7n0=', 'cid': u'14
6637fca78d4ab9a372b254b3b9324b'}
step 5
Authentication successful using "test@moonshot.um.es" moonshot identity.

You have access to the following tenant(s)and domain(s):
  { 0 } swifttenanttest1
Enter the number corresponding to the tenant you want to use: █
```

Thank you for your attention

Any further question?