MTA **SZTAKI**
Hungarian Academy of Sciences
Institute for Computer Science and Control

OpenNebula Multiple AA integration

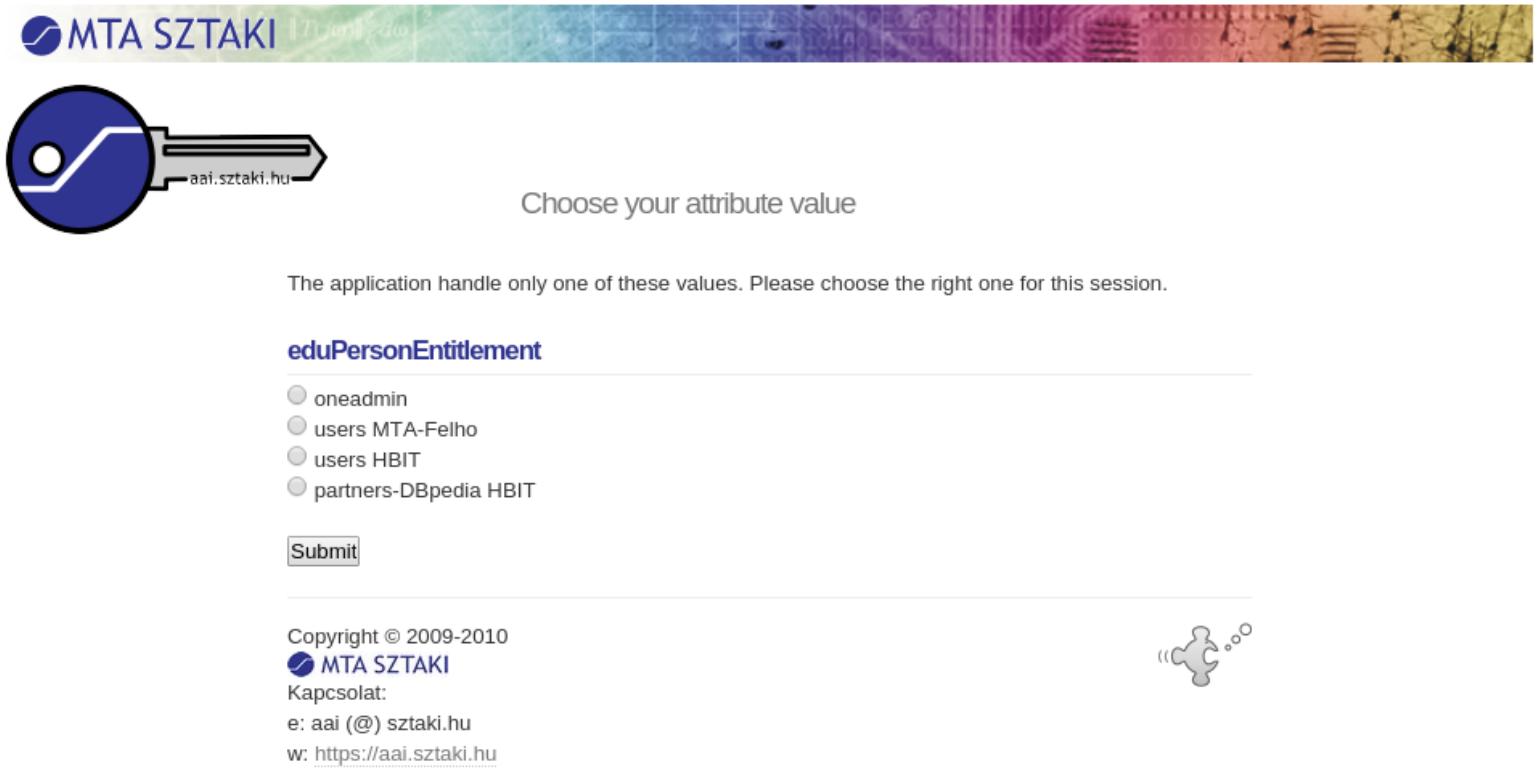
Jan 21 / 2016

Mihály Héder
[pronunciation hint: Mihaay ;)]

OpenNebula history at MTA Sztaki / 1

- SAML+AA integration based on simpleSAMLphp and YAVOM (HEXAA predecessor)
  - In production since 2013
  - OpenNebula 3.6, 3.8, 4.0, 4.2, 4.4
    - *by Milán Unicsovics (MTA SZTAKI), Mihály Héder (MTA SZTAKI)*
  - Custom authorization mechanism specific to YAVOM
  - Contains bugs
  - DEPRECATED

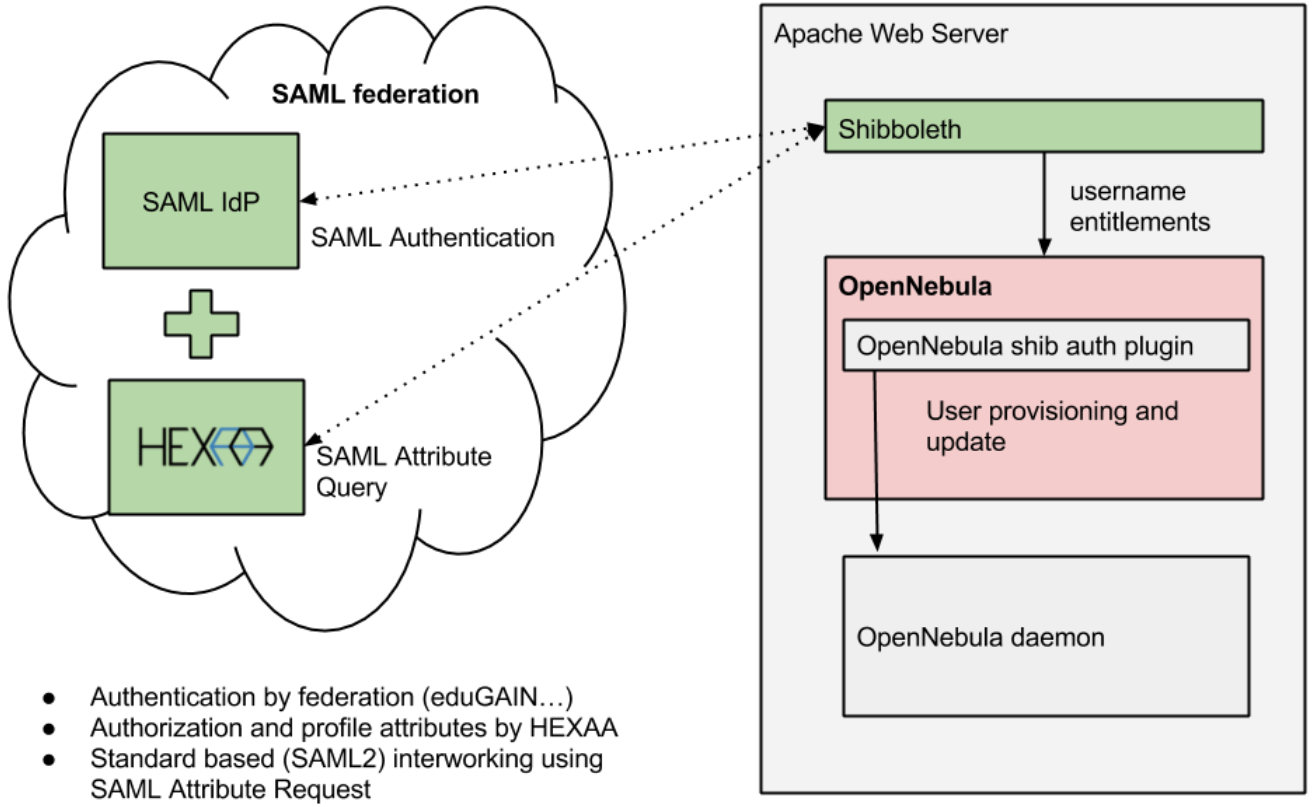
Live Demo of OpenNebula integration 1



OpenNebula history at MTA Sztaki / 2

- SAML+AA integration based on Apache Header variables (we use it whith Shibboleth)
  - In production since 2014
  - OpenNebula 4.4, 4.10.1 (current version is at 4.14.2)
    - *by Milán Unicsovics (MTA SZTAKI), Szabolcs Tenczer (MTA SZTAKI), Mihály Héder (MTA SZTAKI)*
  - Completely relies on apache environment variables
  - From the Federation's point of view it is just any other Shibboleth SP
    - Can be configured with any SAML AA-s, multiple AA-s etc.
  - Uses **primary** and **secondary groups** for authorization
  - Still beta - suspected race conditions
- This approach to SAML integration was the basis of our OpenStack work
  - Héder Mihály, Szabolcs Tenczer, and Andrea Biancini. "Collaboration between SAML Federations and OpenStack Clouds." arXiv preprint arXiv:1510.04017 (2015).

Architecture



- Authentication by federation (eduGAIN…)
- Authorization and profile attributes by HEXAA
- Standard based (SAML2) interworking using SAML Attribute Request

OpenNebula with Shibboleth – Main code

```
#apache config
<Location /one>
AllowOverride all
Order allow,deny
Allow from all
AuthType shibboleth
require valid-use
ShibUseHeaders On
ShibRequireSession On
</Location>

#/etc/one/sunstone-server.conf
:shib_logoutpage: /Shibboleth.sso/Logout --The Shibboleth logout URL
:one_auth_username: oneadmin --Username of the admin user
:one_auth_passwd: oneadminpass --Password of the admin user
:shib_ent_prefix: opennebula --Entitlement prefix
:shib_username: HTTP_EPPN --Shibboleth attribute to use as username
:shib_entitlement: HTTP_ENTITLEMENT --Shibboleth attribute to use as entitlement
:shib_entitlement_priority: --Entitlement priority list
```

OpenNebula with Shibboleth – Main code

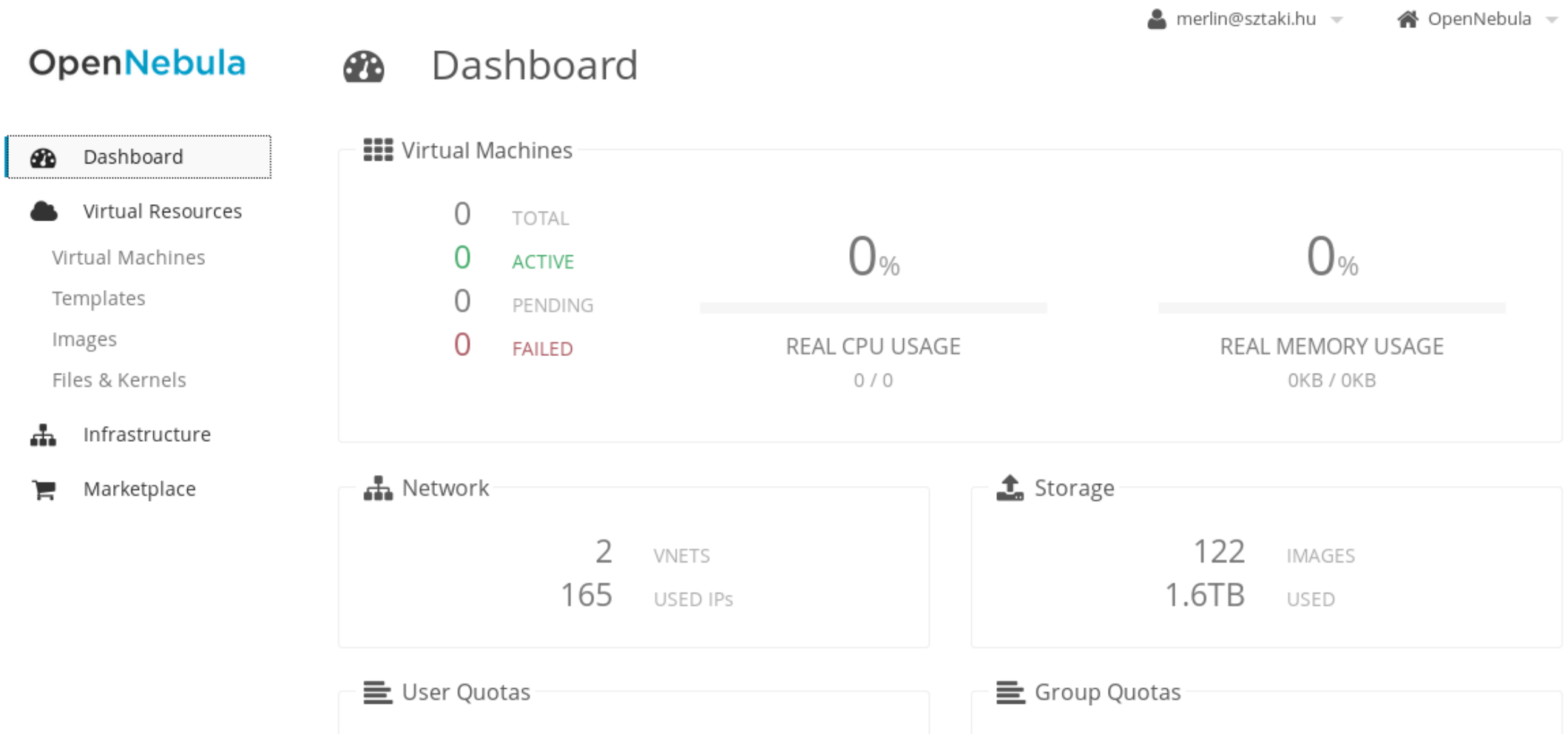
```
# get username from session
username = params['shib_username']

# if new user wants to login then create it
userid = shib.get_userid(username).to_i
if userid == 0
    userid = shib.create_user(username).to_i
end

if !params['shib_entitlement'].empty?
    # get groupnames from entitlement
    groupnames = shib.get_groups(params['shib_entitlement'])
    # add user to given groups remove him from the old groups
    shib.handle_groups(userid, groupnames)
else
    # if new user does not have any entitlement then refuse to login
    return nil
end

return username
```

Live Demo of OpenNebula with Shibboleth



Current Status

- Recent wonderful white paper on OpenNebula+SAML
  - Boris Parak (2015*), Using OpenNebula with SAML-based Authentication and Authorization*, CESNET Author Manuscript
- Issues
  - Authentication and Authorization is not consistently implemented across all interfaces
  - Too tight coupling with subsystems
  - No deprovisioning
  - Patch needs to be maintained currently
- Bottom line: More work needed

Thanks for your attention!
Questions?