H2020-EINFRA-2015-1

VI-SEEM

VRE for regional Interdisciplinary communities in Southeast Europe and the Eastern Mediterranean

Deliverable D3.4

VRE AAI model and compatibility with other e-Infrastructures

Author(s): Peter Molnar (editor)

Status –Version: Final - e

Date: February 5, 2018

Distribution - Type: Internal

Abstract: Deliverable D3.4 – Provides the AAI Model and describes compatibility with other e-Infrastructures.

The VI-SEEM Consortium consists of:

GRNET	Coordinating Contractor	Greece
CYI	Contractor	Cyprus
IICT-BAS	Contractor	Bulgaria
IPB	Contractor	Serbia
NIIF	Contractor	Hungary
UVT	Contractor	Romania
UPT	Contractor	Albania
UNI BL	Contractor	Bosnia-Herzegovina
UKIM	Contractor	FYR of Macedonia
UOM	Contractor	Montenegro
RENAM	Contractor	Moldova (Republic of)
IIAP-NAS-RA	Contractor	Armenia
GRENA	Contractor	Georgia
BA	Contractor	Egypt

IUCC	Contractor	Israel
SESAME	Contractor	Jordan

The VI-SEEM project is funded by the European Commission under the Horizon 2020 e-Infrastructures grant agreement no. 675121.

Document Revision History

Date	Issue	Author/Editor/Contributor	Summary of main changes
December 5th, 2017	a	Peter Molnar	Initial version of ToC
December 8th, 2017	b	Peter Molnar, Tamas Maray	Final version of ToC
January 4th, 2018	c	Anastas Mishev, Dusan Vudragovic, Nicolas Liampotis, Mihajlo Savic, Dimitar Slavov	Contributions from partners added
January 12th, 2018	d	Peter Molnar, Tamas Maray, Lajos Balint	Draft version of deliverable
February 5th, 2018	e	Ognjen Prnjat, Nicolas Liampotis	Final additions and editing

Table of contents

References

[1] Project VI-SEEM-675121 - Annex I - Description of the action

[2] EUDAT2020 https://www.eudat.eu/

[3] ownCloud – https://owncloud.org/

[4] FitSM – standards for lightweight IT service management http://fitsm.itemo.org/

[5] AARC Blueprint Architecture https://aarc-project.eu/architecture/

List of Tables

List of Figures

Glossary

AAI	Authentication and Authorization Infrastructure
API	Application Programing Interface
COmanage	Collaborative Organization Management
DNS	Domain Name System
FitSM	Free Standard for Lightweight IT Service Management
HEXAA	Higher Education eXternal Attribute Authority
IdP	Identity Provider
MDSSO	Multi Domain Single SignOn
NGINX	Engine X - HTTP and reverse proxy server, mail proxy server
OASIS	Advancing Open Standards for the Information Society
REFEDS	Research and Education FEDerations
REST API	Representational State Transfer API
SAML	Security Assertion Markup Language
SP	Service Provider
SPMS	Service Portfolio Management System
SPMT	Service Portfolio Management Tool
SSO	Single SignOn
VHO	Virtual Home Organisation
VI-SEEM	VRE for regional Interdisciplinary communities in Southeast and the
VRE	Virtual Research Environment
WAL	Write Ahead Log

Executive summary

What is the focus of this Deliverable?

Deliverable D3.4 provides the detailed description of the VI-SEEM authentication and authorization infrastructure (AAI), which is based upon international standards and state-of-the-art technologies, and it is integrated with VI-SEEM services to enable an easy and simple, yet secure and well-monitored access to the resources for the registered VI-SEEM users.

What is next in the process to deliver the VI-SEEM results?

The VI-SEEM AAI infrastructure is ready and operational and it is put into production phase. If new services emerge, they can be integrated with the VI-SEEM AAI, to take advantage of the comfortable single-sign-on approach.

What are the deliverable contents?

The deliverable gives an overview of the VI-SEEM AAI approach and justifies the necessity of AAI integration of the services. Then it describes in detail the architecture of the VI-SEEM AAI infrastructure, including the discovery, proxy and authorization modules. It also discusses the compatibility of the VI-SEEM solution with other e-Infrastructures and services. Finally it introduces the AAI integrated VI-SEEM services from the authentication and authorization point of view.

Conclusions and recommendations

The VI-SEEM AAI is a high level, sophisticated and standards-based solution for providing AAI service for those VI-SEEM services that require it. It works efficiently in a multi-domain, distributed, international environment and has reached production level. In order to maintain its high level of reliability, security and performance it is recommended to monitor its operation and to perform regular updates of the main components. The design of the VI-SEEM Login service follows the AARC Blueprint Architecture [5] which is addressing the growing need for research infrastructures and e-Infrastructures to use federated authentication and authorisation mechanisms in ways that will allow interoperation and collaboration.

1. Overview of advantages of AAI integrations

The goal of WP3 is to develop an authentication and authorization infrastructure for VI-SEEM communities that is:

- secure, in order to protect valuable resources;
- distributed, in order to provide a manageable environment for communities with various types and sizes;
- easy to use for the researchers, by facilitating them to access the resources with the credentials they are used to and provide single sign-on where it is possible;
- standards-based, so that the achievements of the work package will be maintainable and interoperable.

Organisations need to manage user information scattered across external and internal application systems.

AAI integrations provide the kind of reliability and accessibility to user access control that is imperative to most project and organisation sites nowadays.

Identity federation is a powerful scheme that links accounts of users maintained distinctly by different organisations. The concept of federated identity is a driver for accelerating automation of Web Services on the Internet for users on their behalf while protecting privacy of their personally identifiable information.

Modern architectures have to control access not only to single, isolated systems, but to whole system of applications and services. This task is complicated by the diversity of today's specifications concerning e.g. privacy, system integrity and distribution in the web.

2. AAI technologies for web resources

Security Assertion Markup Language (SAML) is an XML-based, OASIS data format for exchanging authentication and authorization data between parties, in particular, between an identity provider (IdP) and a service provider (SP). Products supporting SAML version 2 are deployed extensively both at governments, higher education and commercial enterprises worldwide.

Even though the standard was designed to work for all kinds of applications, in practice the Web Browser SSO Profile what is mostly deployed, thus the support for non-web-based applications is problematic.

Over the years, various products have been marketed with the claim of providing support for web-based SSO. These products have typically relied on browser cookies to maintain user authentication state information so that re-authentication is not required each time the web user accesses the system. However, since browser cookies are never transmitted between DNS domains, the authentication state information in the cookies from one domain is never available to another domain. Therefore, these products have typically supported multi-domain SSO (MDSSO) through the use of proprietary mechanisms to pass the authentication state information between the domains. While the use of a single vendor's product may sometimes be viable within a single enterprise, business partners usually have heterogeneous environments that make the use of proprietary protocols impractical for MDSSO. SAML solves the MDSSO problem by providing a standard vendor-independent grammar and protocol for transferring information about a user from one web server to another independent of the server DNS domains.

3. VI-SEEM AAI core components

The VI-SEEM Login service enables research communities to access VI-SEEM e-Infrastructure resources in a user-friendly and secure way. More specifically, VI-SEEM Login allows researchers whose home organizations participate in one of the eduGAIN federations to access the VI-SEEM infrastructure and services using the same credentials they are using at their home organization. Furthermore, VI-SEEM Login supports user authentication with social identities, enabling even those users who do not have a federated account at home organization to be able to seamlessly access the VI-SEEM services without compromising the security of the VI-SEEM infrastructure.

VI-SEEM Login serves as a central hub between federated Identity Providers (IdP) and Service Providers (SP). More specifically, it acts as a Service Provider towards the external Identity Providers and as an Identity Provider towards the VI-SEEM Service Providers. Common technical services, such as Identity Provider discovery and user registration, are provided centrally by VI-SEEM Login, and do not have to be implemented by each individual service. VI-SEEM Login is also responsible for aggregating user attributes originating from various authoritative sources (IdPs and attribute authorities) and delivering them to the connected Service Providers in a harmonised and transparent way. Service Providers can use the received attributes for authorization purposes.

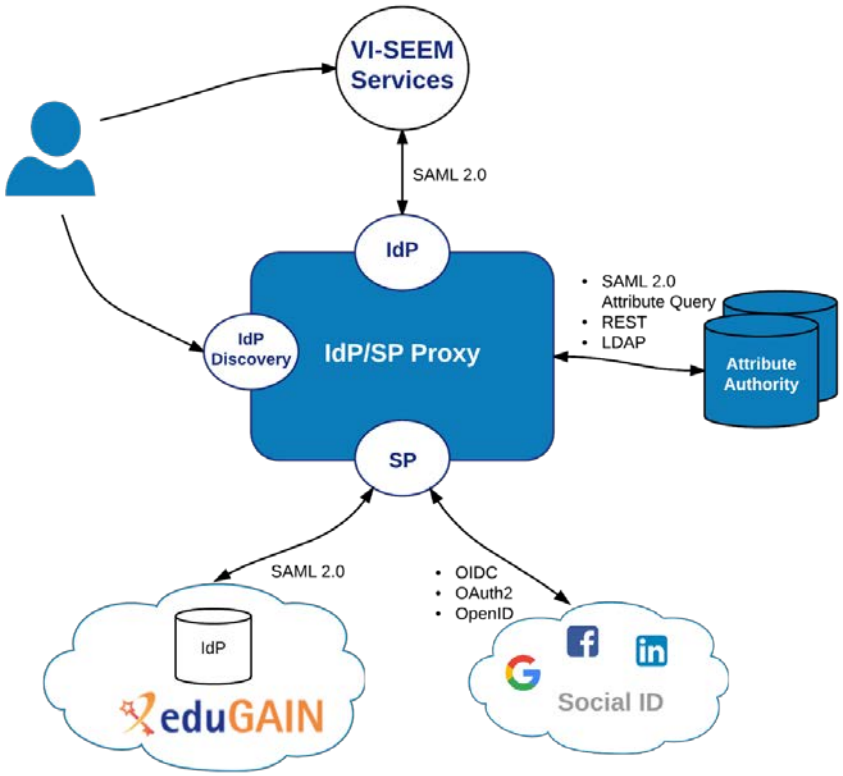

Figure 1: VI-SEEM Login architectural elements

Figure 1 illustrates a high-level view of the VI-SEEM Login architectural elements that deliver the system’s functionality. The view depicts the key functional components, the interfaces they expose, and the interactions between them.

It should be noted that the design of the VI-SEEM Login service follows the AARC Blueprint Architecture (https://aarc-project.eu/architecture/) which is addressing the growing need for research infrastructures and e-Infrastructures to use federated authentication and authorisation mechanisms in ways that will allow interoperation and collaboration.

3.1. VI-SEEM AAI Discovery

The VI-SEEM Login Discovery service is based on DiscoPower, which is an open-source implementation of the Identity Provider Discovery Protocol. DiscoPower is a SimpleSAMLphp module that is part of the official SimpleSAMLphp release. Although not enabled by default, it extends SimpleSAMLphp’s built-in Discovery Service in order to scale to a large number of IdPs. Specifically, DiscoPower supports tabbed organisation of connected IdPs and live search capabilities. It allows users to filter IdP search results as they type (incremental) and is multilingual based on SAML 2.0 metadata information.

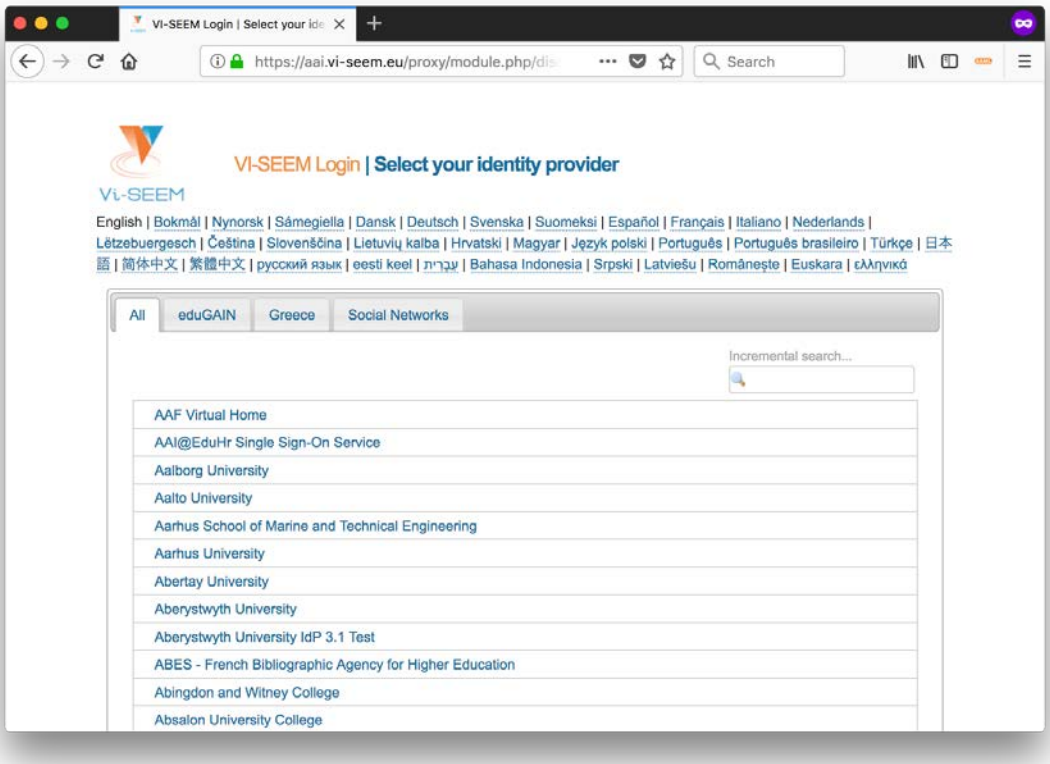

Figure 2: VI-SEEM login discovery page

3.2. VI-SEEM AAI Proxy

The VI-SEEM Login proxy interconnects external Identity Providers (IdPs) with Service Providers (SPs) internal to the VI-SEEM infrastructure. Specifically, the proxy acts as a Service Provider towards the external Identity Providers and, at the same time, as an Identity Provider towards the internal Service Providers (e.g. VI-SEEM Repository, Monitoring, Simple Storage etc). Through the IdP/SP proxy, users are able to sign into VI-SEEM services with the credentials provided by the IdP of their university or research institute that participates in eduGAIN, as well as using social identity providers, or other selected external identity providers, such as Google, Facebook, and LinkedIn. To achieve this, the proxy supports different authentication and authorisation standards, such as SAML 2.0, OpenID Connect 1.0 and OAuth 2.0. The proxy also provides a central Discovery Service (Where Are You From – WAYF) for users to select their preferred IdP (for details please refer to Section 3.1). The core underlying software component of the proxy is SimpleSAMLphp, which is an open-source implementation for federated AAI that supports multiple identity protocols and frameworks, primarily focusing on SAML 2.0.

3.3. VI-SEEM AAI Authorization

3.3.1. COmanage

VI-SEEM Login is using COmanage to handle the initial registration of users with the VI-SEEM infrastructure, as well as the subsequent management of profile information through an intuitive web UI (https://aai.vi-seem.eu/registry). Registration is triggered automatically every time a new user attempts to sign in to any of the federated VI-SEEM services or, explicitly, by visiting https://aai.vi-seem.eu/signup. During the registration process, a user is assigned a personal VI-SEEM ID, which is a persistent, non-reassignable, non-targeted, opaque, and globally unique identifier that allows services to consistently identify users when accessing the VI-SEEM infrastructure. The generated VI-SEEM ID must be accompanied with a set of profile attributes, including the user’s name, email, and affiliation information. The VI-SEEM Login SP Proxy attempts to retrieve these attributes from the user’s IdP; if this is not possible (e.g. due to insufficient attribute release policies), users are requested to supply the missing attribute values themselves during registration (see Figure 3).

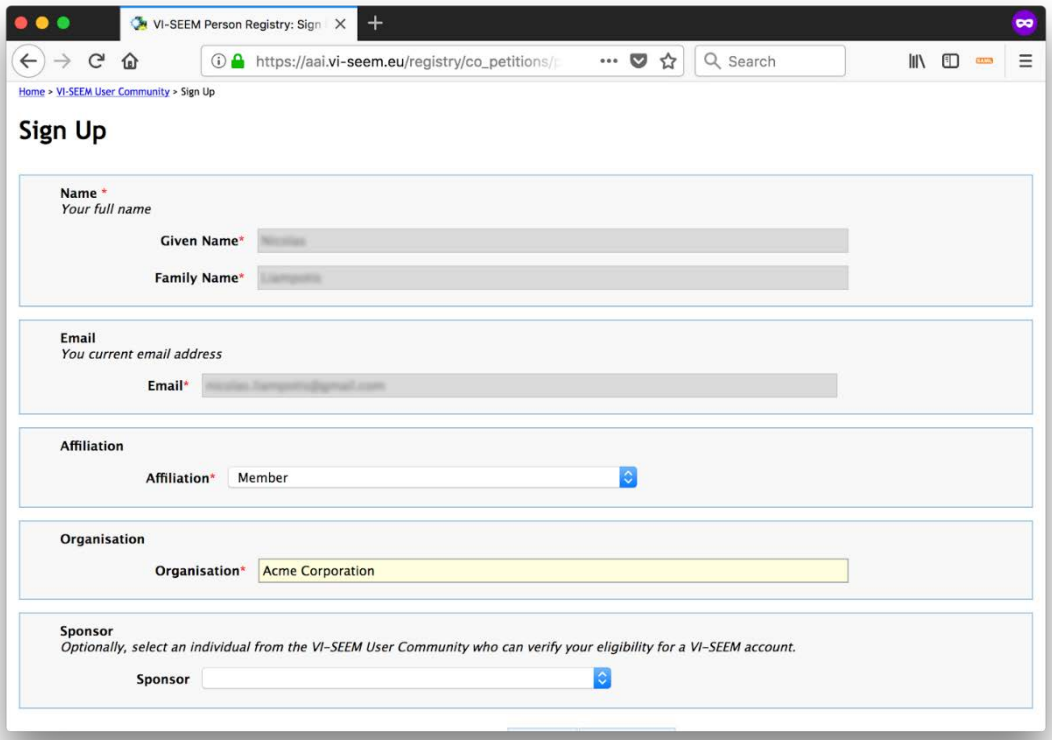

Figure 3: VI-SEEM user registration

The table below describes the information requested from the user’s IdP. As a bare minimum, the IdP is always required to release a long-lasting identifier that uniquely identifies the user within the scope of that IdP. The provided identifier is associated with the personal VI-SEEM ID of the user upon initial registration. The requested information is communicated to the VI-SEEM SP proxy in the form of SAML attributes assertions. Note that the requested set of attributes complies with the REFEDS Research and Scholarship (R&S) attribute bundle, which allows for effective use of services and represents a privacy baseline such that further minimisation achieves no particular benefit.

Description	Required	SAML attribute
Unique long-lasting user identifier (at least one of the following should be released): 1. pseudonymous, non-reassignable identifier 2. name-based identifier 3. pseudonymous identifier	always	1. eduPersonUniqueId 2. eduPersonPrincipalName 3. eduPersonTargetedID or SAML persistent identifier

Preferred name for display purposes, for example to be used in a greeting or a descriptive listing		displayName
First name		givenName
Surname		sn
Email address		mail
Role at Home Organisation		eduPersonScopedAffiliation

Table 1: Information requested from the user’s IdP

After initial registration, COmanage allows users to view their profile information, as well as to link additional institutional or social identities to their personal VI-SEEM ID. Linking enables a user to access VI-SEEM resources with the same VI-SEEM ID, using any of the login credentials they have linked. Figure 4 illustrates the profile information page of a registered user who has linked two different login identifiers (organisational identities) to their unique personal VI-SEEM ID.

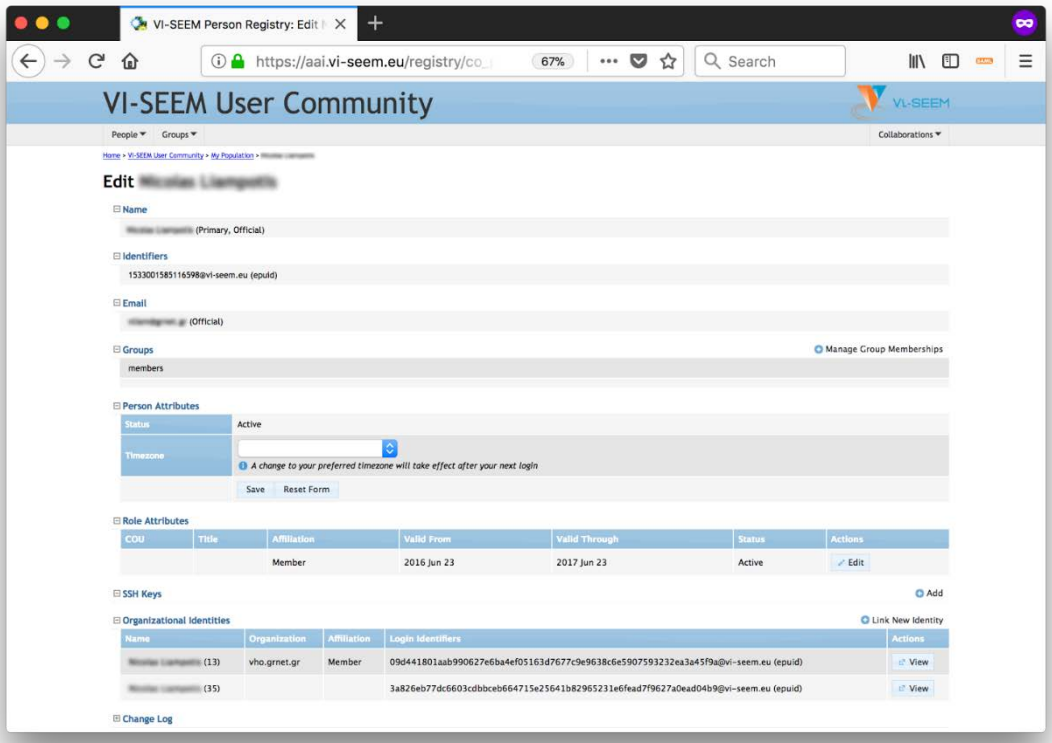

Figure 4: Profile information page

3.3.2. HEXAA

In Higher Education Identity Federations, usually the identities are provided by traditional organizations, such as universities, research institutes, libraries etc. This means that joining such an organization as an individual is normally a formal and well-defined process.

However, some research communities have provided use cases that were stretching the limits of the identity federations. They need information that is relevant to the person’s role within the community and not to the organizational identity, therefore this information should not be managed and provided by the home organization or by the Identity Provider. All the same, the communities still rely on the federations for authentication as a minimum – therefore the trend is to create collaborative environments for identity federations. These environments can be implemented by building external attribute providers that are integrated to existing federations.

HEXAA (Higher Education eXternal Attribute Authority) was designed to be a collaboration platform that implements SAML2 Attribute Authority interface for the Service Providers, thus they can use standard SAML2 Attribute Query protocol to fetch additional information. By doing this, the applications that have already implemented federated authorization can be integrated to HEXAA without any change.

3.4. Virtual Home Organization

While the goal is to use organisation-asserted identities where it is possible, there are a number of researchers in the VI-SEEM project who do not have an existing SAML identity provider to authenticate with. In order to let those individuals to use VI-SEEM services, a Virtual Home Organisation (VHO) must be set up.

VHO is an IdP for sponsored identities. A user might be sponsored either by an individual: every user with an active organisational identity may create sponsored identities; by a community (group): authorised personnel of groups managed in HEXAA might sponsor identities. Groups have more flexibility with sponsoring: custom expiration policy, invitation text, shared administration.

The invited user can choose whether to register a new set of credentials (username and password) or to use his or her existing social identity (Facebook, LinkedIn, Google+) for authentication. Using a social identity might have some benefits, such as no management cost for password management (recover lost passwords, etc), but some drawbacks as well.

3.5. Compatibility with other e-Infrastructures

The growth of Research and Education (R&E) national identity federations and the expansion of eduGAIN, the interfederation service operated by GÈANT, in terms of number of participating countries and entities have demonstrated the increased demand for federated access. The advantage of federated access is that users can

access different services with the same credentials verified by the users’ home institutions in a user-friendly way, while at the same time preserving security and user privacy. Besides eduGAIN and national identity federations, e-Infrastructures and research infrastructures have also deployed Authentication and Authorisation platforms (e.g. EGI Check-in, EUDAT B2ACCESS, ELIXIR AAI, etc.), in order to manage access to their services in a more controlled and consolidated way and also to support the increasing number of use cases requiring federated access to the services provided by different infrastructures. A typical example is accessing storage and computing services provided by different e-infrastructures in a workflow orchestrated by platforms provided by a research infrastructure.

In order to ensure compatibility with the other research and e-Infrastructures, the VI-SEEM Login service is following the AARC blueprint architecture [5]. Specifically, with the adoption of the AARC IdP/SP proxy model, the VI-SEEM Login service takes full advantage of federated identities from eduGAIN and the national identity federations, while reducing the administrative and technical overhead of integrating services within or outside the VI-SEEM infrastructure. The AARC IdP/SP proxy model also facilitates the integration of cross-sector identity services, such as social and other guest identity solutions. Furthermore, the architecture enables the integration with community-managed group management systems to support authorisation policies based on federated identities, augmented by community-specific information. Finally, it should be noted that VI-SEEM Login is following the AARC guidelines on the harmonisation of user attributes, the alignment of the levels of assurance, and the consistent representation of group membership and role information to allow for the interoperable exchange of user and community information among different infrastructures.

3.5.1. eduGAIN connection

As already mentioned, the VI-SEEM Login service supports a number of external authentication providers using various protocols, such as SAML 2.0, OpenID Connect and OAuth 2.0. The different authentication providers are described hereafter.

Integration with eduGAIN allows users to authenticate using the login credentials from their university or research institute. Through eduGAIN, VI-SEEM services that are behind the SP Proxy can become available to more than 2000 IdPs from over 40 Identity Federations with little or no administrative involvement.

3.5.2. Social login

VI-SEEM Login allows researchers who do not have an account at a home organisation federated through eduGAIN to be able to access VI-SEEM services using their existing social media identities. To this end, the following social network providers are supported:

- **Google:** Supports authentication and authorisation through APIs that conform to the OpenID Connect specification. Thus, information about the user is retrieved from the UserInfo endpoint in OpenID Connect format by requesting the openid scope.

- **Facebook:** Authentication and authorisation is based on the OAuth 2.0 protocol. Access to user profile information is provided through the /{user-id} Graph API endpoint, following the OAuth 2.0 login flow.
- **LinkedIn:** Relies on the OAuth 2.0 protocol for enabling authenticated access to its REST APIs that provide access to member data. More specifically, following a three-legged OAuth 2.0 flow, LinkedIn user profile information is accessed through the /people/~ REST API endpoint.

3.6. VI-SEEM AAI login deployment architecture

The components of the VI-SEEM Login service have been deployed on GRNET's ~okeanos cloud infrastructure. As illustrated in Figure 5, an NGINX HTTP server is responsible for load balancing user requests across two instances of SimpleSAMLphp and COmanage. The SimpleSAMLphp instances cache user sessions in Memcached, which serves as an in-memory key-value store for small chunks of arbitrary data. The sessions are distributed and replicated between two distinct Memcached servers, enabling both load balancing and fail-over. The VI-SEEM user profile information managed by COmanage is persisted in a PostgreSQL DB cluster. The cluster consists of a master server, supporting read/write operations, and a hot standby server for read-only queries. The standby is kept current by reading a stream of write-ahead log (WAL) records. If the master server fails, the standby contains all of the data of the master server, and can be quickly made the new master database server.

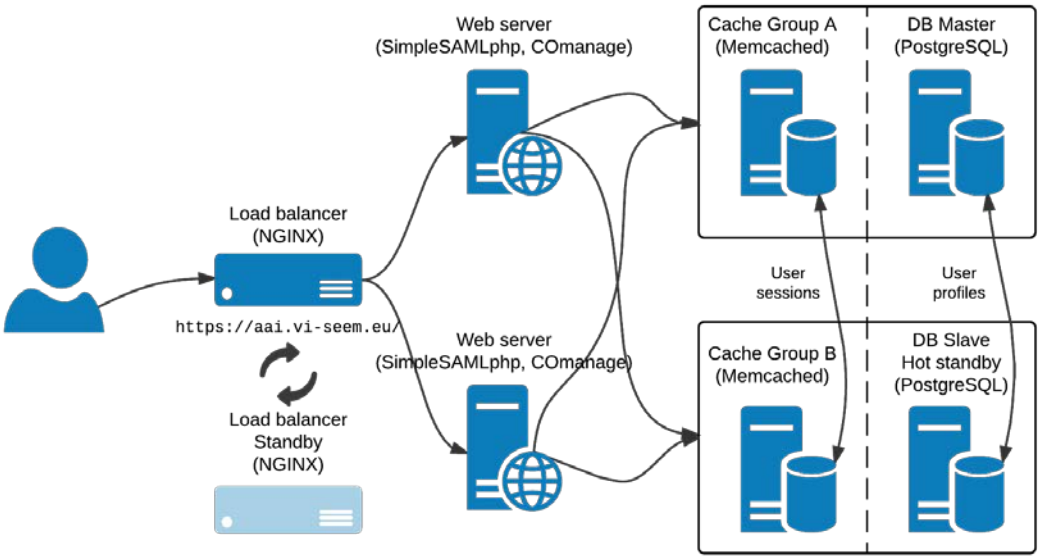

Figure 5: VI-SEEM AAI login architecture

4. VI-SEEM AAI integrated services (SPs)

4.1. Common considerations about VI-SEEM SPs

Federated access to web-based VI-SEEM services is implemented through the use of SAML. The typical Single Sign-On (SSO) flow can be described as follows:

1) The user accesses the VI-SEEM service through their web browser (SP-initiated web SSO).
2) The service redirects the user to the VI-SEEM Login IdP Proxy, asking for authentication (SAML authentication request).
3) The VI-SEEM Login IdP Proxy verifies the SAML request, presents a set of Home Organisations (IdP Discovery Service) to the user and the user selects their preferred Home Organisation.
4) The VI-SEEM Login IdP Proxy generates a new SAML authentication request and redirects the user to the SAML SSO endpoint of the selected IdP.
5) Upon successful authentication, the IdP of the user’s Home organisation sends a SAML authentication response to the VI-SEEM Login SP Proxy.
6) The VI-SEEM Login proxy validates the authentication response and, based on the contained user identifier, it retrieves the profile information stored in COmanage.
7) The VI-SEEM Login proxy builds a new authentication response that encapsulates the retrieved user profile information and sends it to the VI-SEEM service.
8) The VI-SEEM service retrieves the authentication response and validates it.
9) The identity of the user is established and the user is provided with access.

Metadata exchange

SAML authentication relies on the use of metadata. Both parties (the SP and the VI-SEEM Login IdP Proxy) 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.

The format of the exchanged metadata must be based on the XML-based SAML 2.0 specification. Such an XML document can be automatically generated by all major SAML 2.0 SP software solutions (e.g., Shibboleth, SimpleSAMLphp, and mod_auth_mellon). It is important that the metadata are served over HTTPS using a browser-friendly SSL certificate, i.e. issued by a trusted certificate authority. Depending on the software used, the authoritative XML metadata URL for the SP might be in the following form:

- https://example.sp.VI-SEEM.eu/shibboleth (Shibboleth)
- https://example.sp.VI-SEEM.eu/simplesaml/module.php/saml/sp/metadata.php/default-sp (SimpleSAMLphp)

The SAML metadata of the VI-SEEM Login IdP proxy that need to be registered with the SP:

- entityID: https://aai.VI-SEEM.eu/proxy/saml2/idp/metadata.php
- Metadata URL: https://aai.VI-SEEM.eu/proxy/saml2/idp/metadata.php

Attribute release

The VI-SEEM Login IdP Proxy is guaranteed to release the REFEDS R&S attribute bundle to connected Service Providers without administrative involvement, subject to user consent. The bundle consists of the following attributes:

- Persistent, non-reassignable, non-targeted, opaque, globally unique VI-SEEM user ID (eduPersonUniqueId); this is always scoped @VI-SEEM.eu
- Email address (mail)
- Display name (displayName) OR (givenName AND sn)

Affiliation at home organisation (eduPersonScopedAffiliation)

4.2. VI-SEEM Code Repository

For the needs of source code repository VI-SEEM project uses GitLab software. GitLab is comprised of several components that range from local services, over ssh services to web front end. We have integrated web front with the project AAI services by using OmniAuth and using GitLab Omnibus as a starting point.

By default, the configuration is located at /etc/gitlab/gitlab.rb and we need to adapt it to the project infrastructure:

```
gitlab_rails['omniauth_enabled'] = true

gitlab_rails['omniauth_allow_single_sign_on'] = true

gitlab_rails['omniauth_block_auto_created_users'] = false

gitlab_rails['omniauth_auto_link_saml_user'] = true

gitlab_rails['omniauth_providers'] = [

        {

        name: 'saml',

        args: {

                assertion_consumer_service_url: 'https://code.VI-
SEEM.eu/users/auth/saml/callback',
```

```
                idp_cert_fingerprint:
'FA:88:E2:9C:62:A5:C1:2A:0D:39:A4:18:03:D6:5F:0D:9B:EE:DB:56',

                idp_sso_target_url: 'https://aai.VI-SEEM.eu/proxy/saml2/idp/SSOService.php',

                issuer: 'https://code.VI-SEEM.eu',

                name_identifier_format: 'urn:oasis:names:tc:SAML:2.0:nameid-
format:transient',

                },

        label: 'VI-SEEM Login'

        }

]
```

These settings first enable the OmniAuth and SSO operation, while also allowing for auto creating of the users that are not blocked when created. Users are also linked automatically to existing local users if their email address match. This allows for frictionless account creation but can be easily adapted lines 3 and 4.

The issue that can arise in this setup is a result of specific approach to CSRF taken in GitLab and manifests itself as a redirect to login page upon successful AAI login. If this happens and there are lines contaning "Can't verify CSRF token authenticity" in:

/var/log/gitlab/gitlab-rails/production.log

log file, one possible solution is to turn off CSFR in:
/opt/gitlab/embedded/service/gitlab rails/config/environments/production.rb

by setting the following to false:
config.action_controller.allow_forgery_protection = false

If this approach is not suitable, there are other documented solutions in GitLab SAML documentation.

4.3. VI-SEEM Monitoring

In order to integrate monitoring web front at https://mon.VI-SEEM.eu/ with project AAI we have opted to use mod_auth_mellon (https://github.com/UNINETT/mod_auth_mellon) module for Apache web server. This allows us to integrate many different web applications run on such server with minimal changes to the application itself. This approach is, however, practical only in situations when one has the option reconfigure the web server and not just deploy web application on preconfigured server.

Prerequisite for this process is properly functioning web server with configured HTTPS/SSL/TLS and required private key and certificate.

During the integration process we have first obtained the metadata from VI-SEEM AAI service and saved it to /etc/apache2/mellon/idp.xml:

```
/usr/bin/wget https://aai.VI-SEEM.eu/proxy/saml2/idp/metadata.php -O \
/etc/apache2/mellon/idp.xml
```

After that, we configured the server by editing site configuration, for example /etc/apache2/sites-enabled/default-ssl.conf:

```
<VirtualHost _default_:443>

# ... generic setup for virtual host ...

SSLCertificateFile      /etc/ssl/somesite.VI-SEEM.eu.crt

SSLCertificateKeyFile /etc/ssl/somesite.VI-SEEM.eu.pem

# ^ these are the private key and the ceritificate of the server

<Location />

        MellonEnable "info"

        # ^ We want to provide our application with user information

        # if it exists, if not - still run normally


        MellonEndpointPath "/saml2"

        MellonDefaultLoginPath "/"

        MellonSessionLength 8640000

        MellonSPentityId "https://somesite.VI-SEEM.eu/saml2/metadata"

        MellonOrganizationName "en" "VI-SEEM"

        MellonOrganizationDisplayName "en" "VI-SEEM Consortium"

        MellonOrganizationURL "https://VI-SEEM.eu/"


        MellonSPPrivateKeyFile /etc/ssl/somesite.VI-SEEM.eu.pem

        MellonSPCertFile /etc/ssl/somesite.VI-SEEM.eu.crt


        MellonIdPMetadataFile /etc/apache2/mellon/idp.xml
```

```
        MellonIdPPublicKeyFile /etc/ssl/aai.VI-SEEM.eu.crt
        # ^ these are the private key and the ceritificate of the server

        MellonUser "urn:oid:1.3.6.1.4.1.5923.1.1.1.13"

        MellonSetEnv "eduPersonUniqueId" "urn:oid:1.3.6.1.4.1.5923.1.1.1.13"
        MellonSetEnv "eduPersonPrincipalName" "urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
        MellonSetEnv "eduPersonTargetedID" "urn:oid:1.3.6.1.4.1.5923.1.1.1.10"
        MellonSetEnv "displayName" "urn:oid:2.16.840.1.113730.3.1.241"
        MellonSetEnv "givenName" "urn:oid:2.5.4.42"
        MellonSetEnv "sn" "urn:oid:2.5.4.4"
        MellonSetEnv "mail" "urn:oid:0.9.2342.19200300.100.1.3"
        MellonSetEnv "eduPersonScopedAffiliation" "urn:oid:1.3.6.1.4.1.5923.1.1.1.9"
        MellonSetEnv "eduPersonAssurance" "urn:oid:1.3.6.1.4.1.5923.1.1.1.11"

        MellonSamlResponseDump Off
        MellonSessionDump Off
</Location>

<Location /secure>
        AuthType "Mellon"
        MellonEnable "auth"
        # ^ We require the user to log in before giving access
</Location>

<Location /guest>
        AuthType "Mellon"
        MellonEnable "off"
```

```
        # ^ We don't want auth for this directory
</Location>

# ... rest of generic setup ...
</VirtualHost>
```

For the concrete service we have opted to run the whole application in "info" type as we need to provide access to both authenticated and unauthenticated users. The difference is that we limit the amount of data that is presented to the user in following requests.

In order to force logging in we can send the user to "auth" type location or to open "https://someserver.VI-SEEM.eu/saml2/login?ReturnTo=/somedir" URL that will then initiate login procedure on AAI service. The "/saml2" part is dictated by " MellonEndpointPath" in site configuration. Similarly, in order for the user to log out we can initiate the logout procedure by sending the user to "https://someserver.VI-SEEM.eu/saml2/logout?ReturnTo=/somedir".

Since the application is a python application, we had to configure the site like in the example for application located at /var/www/project/application.wsgi:

```
 <VirtualHost _default_:443>
# ... generic settings
   WSGIScriptAlias / /var/www/project/application.wsgi
   <Directory /var/www/project>
        Order deny,allow
        Allow from all
   </Directory>
# ... other settings
 </VirtualHost>
 WSGIPythonPath /var/www/project/
```

There are two documented problems that a service provider might encounter:

1. System clocks out of sync – use NTP or similar approach.
2. Lasso library not supporting SHA256 – install at least version 2.5 or newer.

4.4. VI-SEEM Helpdesk

VI-SEEM project uses osTicket as the software running the https://support.VI-SEEM.eu helpdesk. Since the system must be able to handle tickets related to improperly functioning AAI services as well, we had to preserve the possibility to log in via local username/password pairs as well as via VI-SEEM AAI login for agents handling the tickets.

One possible way to achieve this is by installing and enabling auth-passthru plugin on the osTicket server. This plugin expects the web server to authenticate the user and set REMOTE_USER environment variable to the username of the authenticated user. Due to support for various other external methods, osTicket will strip off anything after the @ character in username, so for user with email like "someone@somesite.edu" osTicket will see just someone as username.

Also important is to set "Authentication Backend" option to "Any Available Backend" when creating the user that can login via the project AAI.

Changes to original mod_auth_mellon configuration described previously are:

MellonEnable "info"

MellonDefaultLoginPath "https://osticket.domain.tld/scp"

MellonUser "urn:oid:0.9.2342.19200300.100.1.3"

MellonSetEnvNoPrefix "REMOTE_USER" "urn:oid:0.9.2342.19200300.100.1.3"

In order to have a link that takes the user to VI-SEEM Login one can put a link to "https://osticket.domain.tld/saml2/login?ReturnTo=/" in desired place in page template (client pages are in include/client and agent pages are in include/staff directories).

In order to enable for logging-out we need to log out twice - from AAI and from local level. This is done by inserting following PHP code in desired page (for example include/staff/header.inc.php):

<a
href="https://osticket.domain.tld/saml2/logout?ReturnTo=https://osticket.domain.tld/scp/logout.php?auth=<?php

echo $ost->getLinkToken();?>"

class="no-pjax"><?php echo __('VI-SEEM Log Out');?>

This has an effect of logging the user out from both AAI and returning the user to local logout URL which will then log the user out locally and redirect to login page.

4.5. VI-SEEM Simple Storage Service

The VI-SEEM Simple Storage service (https://simplestorage.VI-SEEM.eu/) allows project's scientific communities to keep and sync research data on various devices, as well as to share their data, thus making it a useful tool in a collaborative environment. The access is enabled via web browsers, desktop and mobile clients. It is based on the ownCloud platform [3] version 9.0, and deployed on Debian 8 OS machine with 24 Intel Xeon CPU-cores, 64 GB of RAM, and 16 TB of storage space.

In order to integrate the VI-SEEM Simple Storage Service into VI-SEEM AAI, it was configured to act as a SAML Service Provider (SP). In our case, this was done by using the Shibboleth software solution. A dedicated ownCloud application, user_shibboleth, is used to establish Shibboleth authentication for ownCloud users. Relevant Shibboleth packages were installed (in case of Debian OS, libapache2-mod-shib2), and afterwards the service was configured by editing files in /etc/shibboleth folder, namely shibboleth2.xml and attribute-map.xml. The first file is the main configuration file, which contains information about the SP and Shibboleth Identity Provider (IdP, VI-SEEM Login IdP Proxy in VI-SEEM environment). File attribute-map.xml gives instructions to the SP on how to map SAML attributes received from IdP to environment variables used by the VI-SEEM Simple Storage Service. Shibboleth also needs SSL key/certificate pair, which can be generated manually (e.g., by using a command shib-keygen on Debian), or can be browser-friendly certificate and key, as it is a case in VI-SEEM. During the installation, for debugging purposes, Shibboleth SP setup was tested against the TestShib web site (https://www.testshib.org/), which provides means for testing both the Shibboleth SP and IdP services.

After the initial Shibboleth setup, SP metadata were entered manually in the VI-SEEM Login, so that the SP can connect to the IdP Proxy. This is documented at the VI-SEEM Wiki page dedicated to the integration of Service Providers into the VI-SEEM AAI infrastructure http://wiki.VI-SEEM.eu/index.php/VI-SEEM_Login_integration_guide_for_Service_Providers. Such metadata include entityID and Metadata URL, and in the case of the VI-SEEM Simple Storage Service these values are https://simplestorage.VI-SEEM.eu/shibboleth (for entityID) and https://simplestorage.VI-SEEM.eu/Shibboleth.sso/Metadata (for Metadata URL). VI-SEEM Login IdP Proxy metadata are present in the shibboleth2.xml file as well.

The ownCloud app, user_shibboleth (https://github.com/EUDAT-B2DROP/user_shibboleth), was added to the ownCloud-based Simple Storage Service in order to allow authentication via the Shibboleth service. A user_shibboleth folder was added to the ownCloud folder (/var/www/owncloud/apps/), and enabled through the ownCloud Web Interface. Beside of that, an additional Apache location directive is added to the VirtualHost configuration for HTTPS connections, as well as an additional

path in the shibboleth2.xml configuration. Proper attributes released from the VI-SEEM Login IdP Proxy upon successful authentication (e.g., eduPersonUniqueId, mail, display name) are manually enabled in applications' source files since the ownCloud application was originally developed for older versions of ownCloud.

During the integration, several inconsistencies related to the mapping of the attributes coming from the VI-SEEM Login to the Simple Storage Service were found. This was fixed, and a fork of the project has been created at the VI-SEEM source code repository (https://code.VI-SEEM.eu/). The produced VI-SEEM user_shibboleth module is available at: https://code.VI-SEEM.eu/petarj/user_shibboleth.

4.6. VI-SEEM Accounting Portal

The accounting system accumulates and reports utilization of the different types of resources in the infrastructure using standard metrics. It is designed to work with various resources like HPC, Storage, Cloud and any custom type of service that can be defined through some numerical metric. The user can access the information grouped in a table by date, year, country, resource name, virtual research community and application in rows and columns. For each of the resource types there is separate view – compute data for HPC and GRID computing data, cloud data for cloud accounting and storage data for storage accounting. Each generated report is presented in a table and simple charts grouped by the user choice.

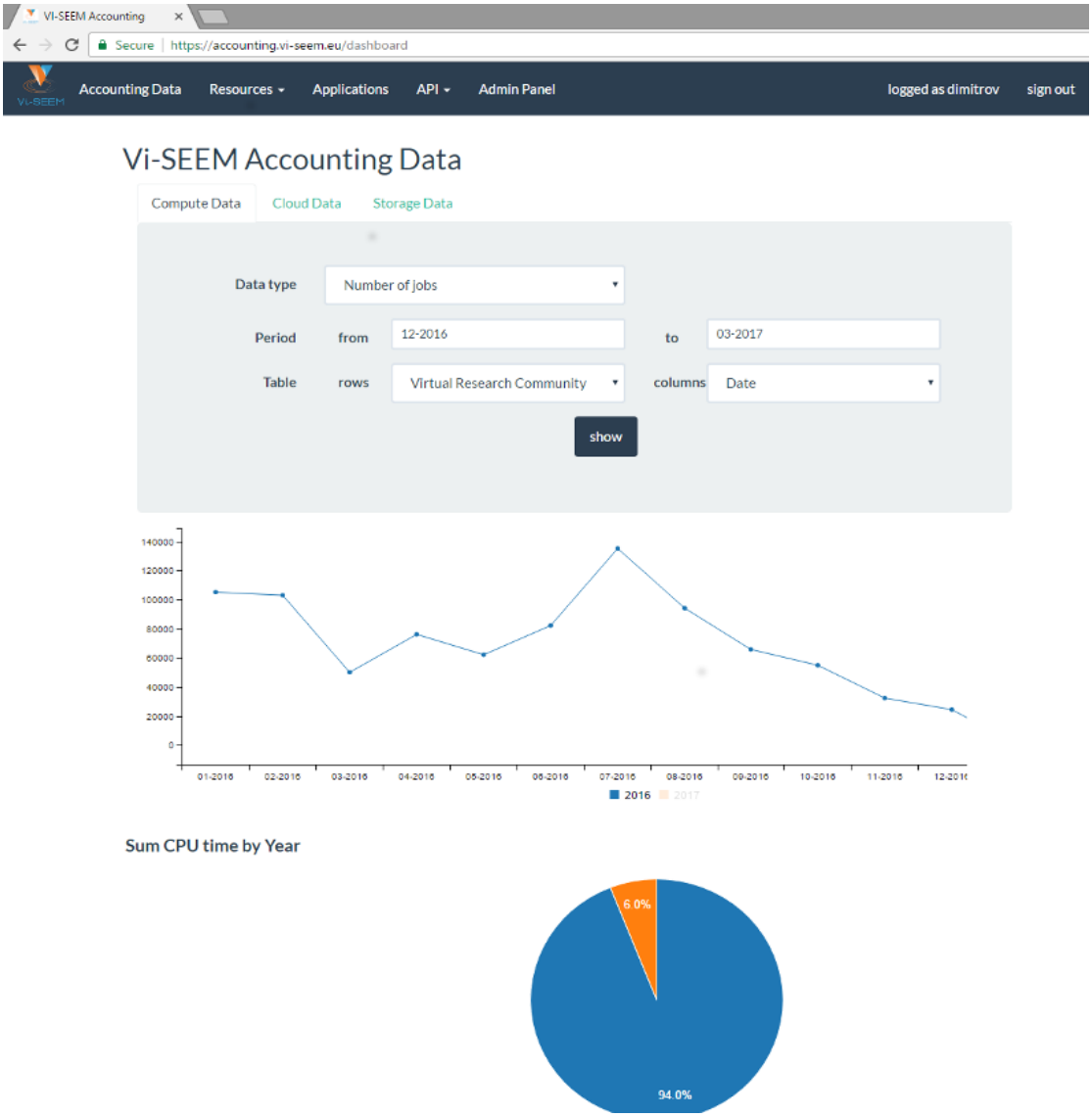

Figure 6: Accounting System UI

The access to the platform is done via web interface that supports the VI-SEEM Single Sign-on mechanism for authentication and authorization. The back-end of the platform is written in python and to integrate the Identity Provider (the AII service) it was used PySAML 2.0 – an open source pure python library for SAML 2.0 protocol.

The user roles in the accounting system have 3 levels of hierarchy – regular user, site manager, and administrator. The regular user role allows the user to browse and extract reports about the usage and can be limited to the information he can see. The site manager role is for the local admins where they have extended rights to configure the resources they manager – to describe an application in their resource, to map HPC queue directly to application to retrieve an API key the REST API and so on. The administrator of the accounting system has all the privileges above and one extra tab view where a summary of the published log records is shown and a resource management option to update user privileges.

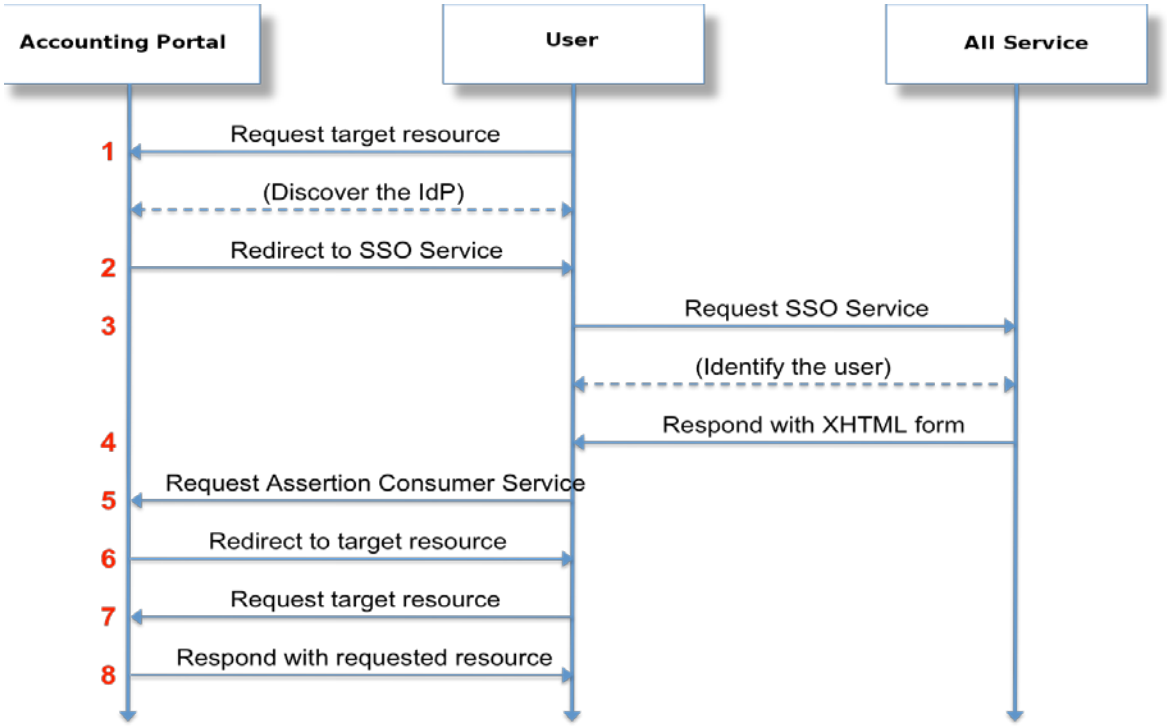

Figure 7: SAML Authentication Mechanism

4.7. VI-SEEM Service Portfolio Management System

The VI-SEEM service portfolio management system has been developed to support the service portfolio management process within VI-SEEM as well as being usable for other infrastructures if required. The main requirements for the creation of this tool have been collected from the service management process design within VI-SEEM work packages WP3 (infrastructure services), WP4 (storage services) and WP5 (application level services). The service management system has been designed to be compatible with the FitSM [4] service portfolio management. Requirements gathered in the context of EUDAT2020 [2] project have also been considered for compatibility and completeness.

The roles incorporated in SPMS are

- The potential customers or end users of the services that are listed in the service catalogue. Such users should be able to see the list and details of the services that are currently into production or beta stages and are for offer to them. Such users should be also able to order or use the services via the service catalogue, interact with the helpdesk or the dedicated support channel for that service and see features and use cases of each service.
- The service managers within the VI-SEEM environment. Such users should be able to see all the details for the services that are in the service portfolio

which contains a superset of the service and the information found in the service catalogue.

- The service owners that are the persons responsible for each service listed in the service portfolio. Members with this role have the full responsibility of the content that is provided within the service catalogue and portfolio for the services under their responsibility.

Data access models

The VI-SEEM catalog/portfolio management system currently allows three different data entry procedures: via the RESTful API, via the temporary service administration tool (SPMT WRITE DJANGO UI) and write UI.

The SPMT READ user interfaces are available under the following links:

- The Service Catalogue View UI: https://services.VI-SEEM.eu/ui/catalogue/services/
- The Service Portfolio View UI: https://services.VI-SEEM.eu/ui/portfolio/services/

The Service Administration interface (SMPT WRITE UI) can be access via https://services.VI-SEEM.eu/ui/service, while the SPMT WRITE DJANGO UI is available at https://services.VI-SEEM.eu/api/admin

Authentication and Authorization

Authentication to the SPMT WRITE UI is handled via VI-SEEM AAI. Anyone with a VI-SEEM AAI account can create an account in the SPMT WRITE UI. However, access rights deny any write operation in the SPMT.

Authorization in SPMT is handled via the tool itself. A user has access rights to the tables of the SPMT database only if it is authenticated by the SPMT WRITE UI administrator. In V1 of the SPMT WRITE UI every user account that is registered as service owner will have write permission to all data of existing services, as well as the right to create new services. In future versions of the tool fine great authorization will prevent a service owner from editing services that do not belong to him.

AAI Integration

The system is also integrated with the VI-SEEM AAI system implemented through SAML authentication. Every user that can authenticate through the VI-SEEM Identity Provider can also use and manage the whole application.

DjangoSAML2 – SAML2 authentication package for Django

Djangosaml2 is a Django application that integrates the PySAML2 library into your project. This means that we can protect the Django based project with a service provider based on PySAML. This way we can use a SAML Identity Provider to control the auth process and return a user ready to be processed by our application.

Since users created in the standard Django Auth module are different than the user objects provided by the AAI system, the user objects In this applications are hybrids containing properties from both types. For the attributes originating from the AAI system there is a mapping that allows for proper user object creation. In case a user has been made with the application’s standalone user module, a reverse mapping has been designed that can forward attributes to link to a AAI system user.

We modified the DjangoSAML2 package to not only serve the Authentication pipeline, but also the Authorization pipeline by allowing direct access to the Roles-Permissions table. This way the overhead of initializing different modules on user login/ creation/ modification/ logout is reduced to a minimum and administrators and moderators would be able to add and remove permissions on the fly.

An additional modification is the notification mailer system for the AAI module. Every major action (creating a user, removing a user, blocking a user) is announced to system admins by an email in the moment of execution. The email contains the necessary information to identify the origins and target of the command as well as the information necessary to recover from potentially malicious actions.

Our AAI flow is enforced through the use of modified decorator functions e.g. loginrequired() (from the Django backend). This function can be appended to any controller and act as a filter for unsigned on unauthorized requests. The modification allow it to forward the request to the hybrid user module where the request is processed according to its nature – with the native pipeline (a native application user) or the AAI pipeline (remotely created user).

Django Social Auth - Package that provides user registration and login using social sites credentials

Our SocialAuth engine is ready to handle any OAuth2 authentication scheme. For OAuth we are limited to the library’s proprietary methods, however it can be extended to serve more platforms. As with the AAI system users, all the users created by the SocialAuth package are mapped to the native user module of the system so all the attributes and functions are enabled. This also means that a SocialAuth user can be mapped to an user from the AAI system and be managed by an external backend.

Registration and Login using social sites using the following providers:

- Google OpenID
- Google OAuth
- Google OAuth2
- Yahoo OpenID
- OpenId like myOpenID
- Twitter OAuth
- Facebook OAuth
- Amazon OAuth2

5. Conclusion

Section 1 presented, on the basis of the VI-SEEM WP3 investigations, important information about the advantages of AAI integration. The brief overview didn’t enter into much detail but rather did draw the attention on key benefits of integrated AAI solutions regarding reliability and accessibility to user access control.

Section 2 has been devoted to AAI technologies for web resources. In this segment of Deliverable D3.4 information can be found about the issues of exchanging authentication and authorization data between an identity provider and a service provider, with special emphasis on web based applications.

Section 3 is dealing with the VI-SEEM AAI Core components. Among these Core items the AAI Discovery, AAI Proxy, and AAI Authorization (COmanage, HEXAA) are separately dealt with, together with Virtual Home Organization, compatibility with other initiatives (including EduGAIN connection and Social login), and AAI login deployment architecture aspects. All these Core components and aspects do play important role in the VI-SEEM AAI solution.

Section 4 provides a treatise summary for VI-SEEM AAI integrated services. After introducing the common considerations about VI-SEEM SPs, description of the VI-SEEM Repository and the Code Repository follows. Next, Monitoring, the Simple Storage Service, and the Accounting Portal aspects are outlined with respect to VI-SEEM e-Infrastructure applications. Finally, the VI-SEEM Service Portfolio Management System is introduced, by briefly referring to the roles incorporated in SPMS, the data access models, the user interfaces, to the handling of authentication and authorization, as well as to AAI integration.

