

21.05.08

Deliverable DJ5.2.3,3: Best Practice Guide - AAI Cookbook - Third Edition

Guidelines for Connecting to the eduGAIN AA
Infrastructure



Deliverable DJ5.2.3,3

Contractual Date:	28/02/08
Actual Date:	21/05/08
Contract Number:	511082
Instrument type:	Integrated Infrastructure Initiative (I3)
Activity:	JRA5
Work Item:	5
Nature of Deliverable:	R (Report)
Dissemination Level	PU
Lead Partner	RedIRIS
Document Code	GN2-08-130

Authors: Diego R. Lopez (RedIRIS), José-Manuel Macías (RedIRIS), Maurizio Molina (DANTE), Jürgen Rauschenbach (DFN), Andreas Åkre Solberg (UNINETT), Manuela Stanica (DFN), GN2 JRA5 team

Abstract

This deliverable introduces the main concepts, protocols and profiles of the GÉANT Authentication and Authorisation Infrastructure (eduGAIN), explains the trust model employed, and provides guidelines on how to connect an already existent AAI to eduGAIN.

Table of Contents

0	Executive Summary	iv
1	Introduction	1
	1.1 Notation	1
2	eduGAIN Architecture and Components	2
	2.1 Component Description	3
	2.1.1 Bridging Elements (BE)	3
	2.1.2 Federation Peering Points (FPP)	4
	2.1.3 Metadata Service (MDS)	4
	2.2 Component Identifiers	4
	2.2.1 Examples	5
	2.3 The eduGAIN Naming Registry	5
	2.4 The eduGAIN Schema	6
3	The eduGAIN Protocols and Profiles	7
	3.1 eduGAIN Basic Profile	8
	3.2 Publishing and Retrieving Metadata	8
	3.3 Web Single Sign On (SSO) Profile	9
	3.4 Automated Client Profile	10
	3.5 User behind a Client Profile	10
	3.6 Client in Web Container Profile	11
4	The eduGAIN Trust Fabric	13
	4.1 PKI Structure	13
	4.2 Trust Validation Procedures	14
5	Software for Connecting to eduGAIN	16
	5.1 The eduGAIN API	16
6	A Checklist for Connecting to eduGAIN	18
7	Conclusions	19
8	References	20

Table of Figures

Figure 2.1: Using the eduGAIN components and protocols to establish trust links across federation limits	3
Figure 3.1: Schematic overview of an abstract eduGAIN operation	7
Figure 3.2: Message flow in the WebSSO profile	9
Figure 3.3: Message flow in the AC profile	10
Figure 3.4: Message flow in the UbC profile	11
Figure 3.5: Message flow in the WE profile	11
Figure 4.1: Structure of the eduGAIN PKI	14
Figure 5.1: Layer diagram for the eduGAIN API	17

0 Executive Summary

This deliverable summarises the main concepts, protocols and profiles of the GÉANT Authentication and Authorisation Infrastructure (eduGAIN) and explains the trust model used. Equipped with this knowledge, the reader will find guidelines for the steps necessary to connect an already operational Authentication and Authorization Infrastructure (AAI) to eduGAIN.

This document is not intended as an AAI or eduGAIN primer, but as a guide for AAI administrators willing to participate in eduGAIN. Since federated services are moving more and more into the centre of interest in many countries, it is important to be informed from the beginning about how such national federation developments can be integrated into the international cooperative environment.

Though eduGAIN has not yet reached the level of production service, it is currently a stable infrastructure connecting more than ten European national identity federations, providing service to several distributed multi-domain applications, and running test interconnections with other federated identity services worldwide

The eduGAIN basic concepts are introduced in the beginning of this document (a more detailed presentation can be found in the “GÉANT2 AAI Architecture” document DJ5.2.2 [GN2DJ522]), followed by a description of the role and functioning of the architecture components and the naming conventions used to designate them. This set of naming conventions, together with the Metadata Service (MDS) and the Public Key Infrastructure (PKI), is vital to ensuring safe and trustworthy communication between the resource owner and the users’ home institution belonging to different local federations. A description of the metadata interactions (publishing and retrieval) necessary in this process is provided further on in the document.

The currently implemented eduGAIN protocols and profiles are also presented so as to provide a better understanding of their functioning in practice. Each profile is defined as the precise exchange of messages and the processing rules for the messages in a particular use case. An essential part of the eduGAIN functionality is its trust model, which needs to be thoroughly understood and applied within each participating federation. Therefore, the different elements constituting the eduGAIN trust fabric are described, including the validation strategies for connections and signatures that must be followed. Finally, the roadmap for connecting to eduGAIN provides an overview of the necessary steps to be taken for joining the confederation.

This document, also named “The eduGAIN Cookbook” (to indicate its main purpose) contains only technological guidelines. The eduGAIN confederation policy will be covered in a separate document in a later project phase.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

1 Introduction

eduGAIN is the confederation technology¹ developed by the GÉANT2 project in order to achieve the interconnection of federated Authentication and Authorisation Infrastructures (AAI). This document provides a series of guidelines for connecting a given national AAI to eduGAIN. It is intended to collect and provide, in a concise manner, the information available about the eduGAIN technological procedures and requirements, which are currently scattered throughout the rest of the eduGAIN technical documentation and code.

It is important to note that this version provides only technological guidelines. Organisational issues will be addressed in a separate document, covering the eduGAIN confederation policy.

1.1 Notation

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [RFC 2119]:

... they **MUST** only be used where it is actually required for interoperation or to limit behaviour which has potential for causing harm (e.g., limiting retransmissions) ...

These keywords are thus capitalised when used to unambiguously specify requirements over protocol and application features and behaviour that affect the interoperability and security of implementations. When these words are not capitalised, they are meant in their natural language sense.

Example code and listings of XML schemas appear like this.

¹ Although not fully internationally standardised, the term “confederation” is more and more used to refer to infrastructures allowing different federated AAIs interoperability. The eduGAIN group has been actively promoting its use as this reflects the way in which federation interconnection is achieved using similar principles

2 eduGAIN Architecture and Components

eduGAIN is an authentication and authorisation infrastructure (AAI) based on the **confederation** concept. As a confederation, eduGAIN provides the means to interconnect a set of national or community-wide federated AAls. These participating federations cooperate to provide services to their member organisations and users beyond their limits. The confederation requires that both identity management and authentication/authorisation services are properly handled by the participating federations, as it only provides the means to enable their interoperation.

Since members of a participant federation do not know in advance about members in other federations, a procedure to establish trust among them is required. Trust links are established by means of a common trusted source for metadata (the Metadata Service , MDS), and used by specific eduGAIN components (the Bridging Elements, BEs), that perform the appropriate adaptation between the eduGAIN and the local federation trust environments. Metadata about a certain federation as a whole are maintained by its Federation Peering Point, though other components (BEs) that the MDS recognises as authoritative sources for metadata can perform partial updates, according to the participant federation wishes.

eduGAIN establishes trust through a **Public Key Infrastructure** (PKI), and a set of **naming conventions** for its components. The relevant information about the eduGAIN components is stored at the MDS and dynamically retrieved and updated via a metadata exchange interface based on the **REST** (Representational State Transfer [REST]) architecture model. Exchange of security information between components is enhanced by the use of the XML-based OASIS standard **SAML** (Security Assertion Markup Language).

Figure 2.1 shows how the eduGAIN components and protocols (in green) are used to establish trust links among resources and identity repositories inside different participating federations, without affecting their local procedures and protocols, shown in different colours for each of the two federations in the diagram. The “H” and “R” prefixes used in the figure are thoroughly employed in eduGAIN to designate the components at the “Home” federation (the one providing user attributes in a particular interaction) and the “Remote” federation (the one controlling the access for a resource in a particular interaction).

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

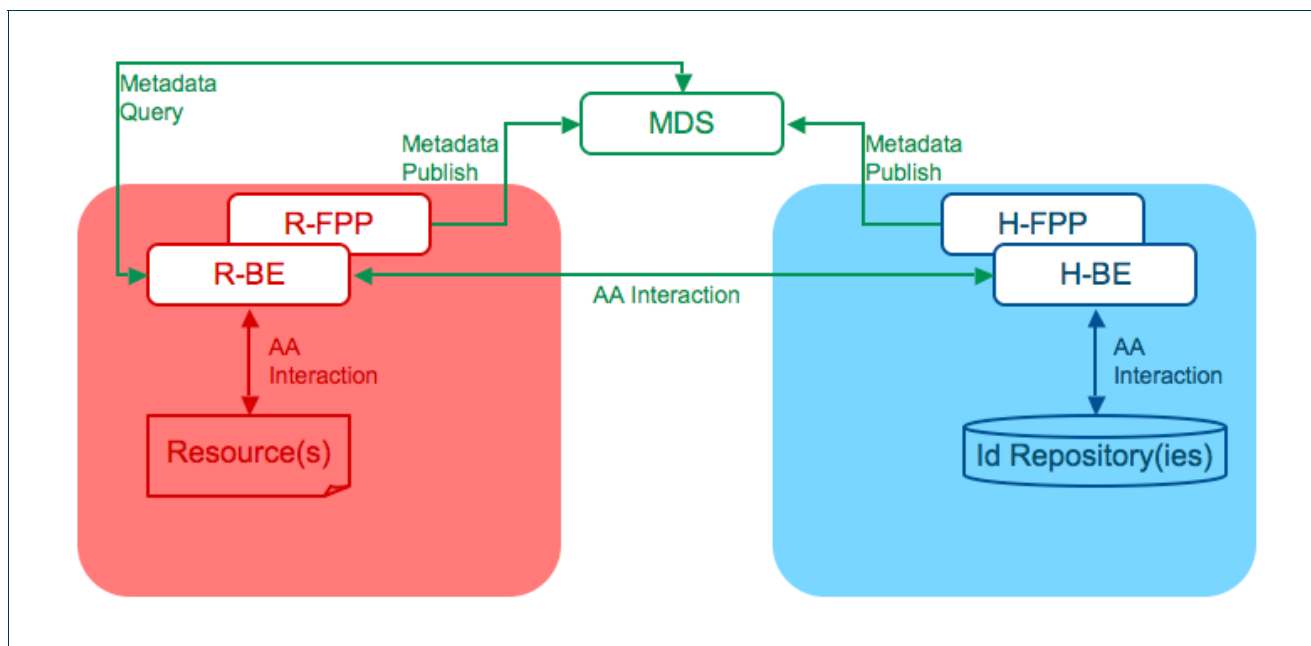


Figure 2.1: Using the eduGAIN components and protocols to establish trust links across federation limits

As the figure above shows, the eduGAIN architecture consists of three main types of components: **Bridging Elements (BE)**, **Federation Peering Points (FPP)** and the **Metadata Service (MDS)**. Their interactions contribute to establishing the trust among the participating federations.

2.1 Component Description

2.1.1 Bridging Elements (BE)

BEs are always integrated within a participating federation and serve as a means of establishing appropriate trust links among federation components and user applications, and of adapting syntax, semantics and procedures used by the participating federations. They are the objects (and subjects) of trust when crossing federation limits. The eduGAIN trust will be maintained among these elements (as they are eduGAIN-aware, the MDS will know about them). The internal trust of each BE with respect to its local federation must be established according to the appropriate local procedures. This transforms the problem of maintaining an NxM trust matrix problem into a one-to-one mapping from eduGAIN trust into the corresponding internal federation trust.

There can be one single BE in a federation, acting as a trust aggregator for the other AAI elements in the local federation, or multiple BEs. The first case is that of a **Local Federation Adaptor (LFA)**, generally corresponding to the case where established infrastructures are in place but the AAI elements are not yet eduGAIN aware. Here a LFA is used to adapt the established infrastructure's own protocols/profiles/procedures to the eduGAIN interfaces and serves as a means of aggregating trust for the whole federation.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

In time, as local federation software becomes eduGAIN-aware, it is expected that individual components of the established infrastructure will be allowed to interact directly within eduGAIN by means of BEs called **Local Adaptors** (LA). These are very similar to LFAs except that they do not aggregate trust for the entire federation, and may in fact even interface to a single host.

2.1.2 Federation Peering Points (FPP)

FPPs serve as a means of publishing metadata about a federation through the MDS (see below). For each federation connected to eduGAIN there is exactly one FPP, which SHOULD be dynamically informed of the state and changes of all BEs within its federation. The FPP thus plays the role of a central administration system by means of which each federation can announce its practices via the MDS and keep other participants informed of changes.

2.1.3 Metadata Service (MDS)

The MDS serves as a means of storing and providing metadata about eduGAIN interfaces, such as identity providers (IdP), Attribute Authorities (AA), Service Providers (SP) and others. Its main use consists of locating the appropriate identity providers able to identify a certain entity in a given federation.

For this purpose, the FPP or authorised BEs within a federation publish the relevant metadata, related to the available local interfaces, at the MDS. Upon inquiry from other BEs, this metadata can then be dynamically retrieved during the trust establishment process by means of HTTP query/response exchanges taking place via the REST interface between BEs and the MDS.

2.2 Component Identifiers

Since the MDS serves as a means of acting as an authoritative and trusted source of metadata among otherwise mutually unaware federations, a way for uniquely identifying a certain element within the whole eduGAIN fabric is required. Neither identity nor service providers at each participating federation have direct access to the certificates used during peer validation. They need to establish a dynamic trust link through the BE and the trust anchors² exchanged via the MDS. The trust validation process is obviously enhanced (both in its processing and in its further auditing) by using identifiers with a formal, well-established format.

Therefore, all components in eduGAIN message elements and assertions MUST be identified according to the following rules:

- Identifiers SHALL be coded by means of URNs in the `urn:geant:edugain:component` namespace.

² The elements where the evaluation of the level of trust on a connecting component is started at. In PKI-based trust schemas, trust anchors are typically the self-signed certificate(s) of the trusted root CA(s).

- Identifiers SHALL establish the kind of component they apply to by means of the following predefined prefixes:
 - `urn:geant:edugain:component:mds` for a Metadata Server.
 - `urn:geant:edugain:component:fpp` for a Federation Peering Point.
 - `urn:geant:edugain:component:be` for a Bridging Element.
 - `urn:geant:edugain:component:sp` for a Service Provider.
 - `urn:geant:edugain:component:idp` for an Identity Provider.

This list is only indicative. The exact relation of valid identifier prefixes MUST be retrieved from the eduGAIN name registry at <https://registry.edugain.org/>

- Identifiers SHALL follow the hierarchy of the trust establishing process, up to the identifier of the participating federation.

Other naming schemas MAY be considered as acceptable for identifying eduGAIN components, as long as they fulfil the requirements of uniqueness and appropriate registration. These candidate schemas MUST be accepted by the eduGAIN confederation participants prior to their inclusion in the eduGAIN deployment procedures.

2.2.1 Examples

A typical MDS identifier should be like:

```
urn:geant:edugain:component:mds:galaxian
```

A typical FPP identifier should be like:

```
urn:geant:edugain:component:fpp:starfleet
```

A typical BE identifier should be like:

```
urn:geant:edugain:component:be:starfleet:enterprise
```

2.3 The eduGAIN Naming Registry

The proper management of the identifiers described above, as well as many other elements in the eduGAIN protocols and infrastructure (protocol components, attribute references, well-defined attribute values, etc.) requires the existence of an **eduGAIN Naming Registry**, serving as a means of publishing and maintaining namespace allocations. The eduGAIN Naming Registry is available at the following URL: <http://registry.edugain.org/>

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

This registry operates the `urn:geant:edugain` namespace, by direct delegation from the `urn:geant` registry. It contains the branches and final values acceptable inside the namespace, including the corresponding delegations where applicable. The namespace values are accessible by a Web interface, providing human-readable HTML pages to be used as reference. The registry offers other interfaces, more oriented towards direct machine access.

Individual identifiers, as well as delegation of whole branches of the managed namespace, can be requested through the registry Web pages. In the former case, the registry operators SHALL assess the legitimacy of the identifier request prior to satisfy it. In the latter, after the request has been signalled as legitimate, a further check on the availability of appropriate interfaces at the delegation point will be performed. Servers holding namespace delegation MUST provide Web pages and Web services that are functionally equivalent to the ones provided by the eduGAIN Naming Registry.

The only identifiers acceptable to an eduGAIN infrastructure element MUST be those included or directly derived from this registry, which is the only reference for eduGAIN software development and deployment. Elements issuing to be accepted within eduGAIN SHALL verify the legitimacy of the eduGAIN identifiers claimed, and SHOULD use for this purpose the machine readable interface to the eduGAIN registry.

Should other namespaces become acceptable in the future (as described in the previous section), there SHALL be an explicit reference to them at the eduGAIN Naming Registry.

2.4 The eduGAIN Schema

Attributes exchanged by the eduGAIN components SHALL be in accordance to the SCHAC schema [SCHAC]. SCHAC stands for SCHEMA for ACademia and provides a set of attributes, agreed among the European NRENs, for the exchange of person and institution related information and not yet covered by previously defined schemas. Any application using eduGAIN will be able to select the appropriate subset of SCHAC attributes. It is important to note that when talking about "SCHAC attributes" this refers to the whole set of them as defined by SCHAC; that is, not just SCHAC-specific attributes (identified by the `schac` prefix in their identifier), but also those defined by the schemas that the SCHAC document assumes are available and properly coded.

Metadata documents about eduGAIN components SHOULD include the appropriate references to the attributes that are requested and/or asserted. For this purpose, they MUST use the attribute identifiers defined by the corresponding standards and registries.

3 The eduGAIN Protocols and Profiles

In the purpose of performing authentication and authorisation interactions, the Security Assertion Mark-up Language (SAML), in combination with SOAP transport over a secured channel, is in wide use and already provides large parts of the required functionality. The figure below provides a schematic overview of a general eduGAIN interaction.

SAML is a set of standards well suited to the eduGAIN tasks. Actual messages to be exchanged by the eduGAIN elements consist essentially of variants of the SAML messages. The SAML data type definitions used in this document correspond both to versions 1.1 [SAML11] and version 2 [SAML20], according to the evolution of the current federation technologies along the lifetime of the project.

As eduGAIN evolves, other profiles covering additional use cases are likely to arise. These profiles SHALL be appropriately documented by a detailed profile specification and an update of this document.

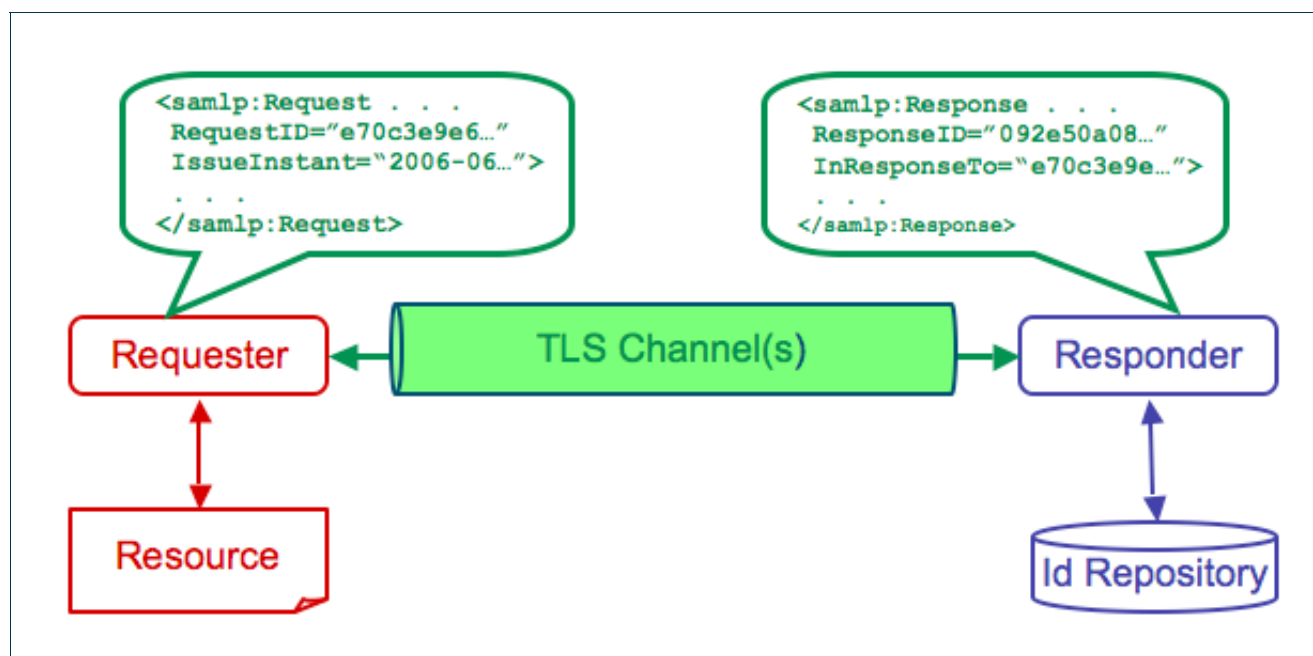


Figure 3.1: Schematic overview of an abstract eduGAIN operation

3.1 eduGAIN Basic Profile

This profile SHALL be the default profile to be used for access to the eduGAIN service definition. It consists of a direct mapping (almost) of the eduGAIN abstract service definition onto SAML over SOAP/HTTP/TLS channel, with the exception of the Metadata Service, which follows a specific profile described in the following section.

eduGAIN supports two different mappings, for SAML 1.1 and SAML 2.0. For backwards compatibility reasons, the SAML 1.1 SHALL be considered the default one, until a complete deployment of the SAML 2.0 eduGAIN infrastructure is achieved.

Both mappings are based on the same general rules:

1. All **Requests** are conveyed as SAML `Request` elements.
2. All **Responses** are conveyed as SAML `Response` elements.
3. Error responses are distinguished from other responses by means of their main `StatusCode` element.

A detailed description of this profile and its mappings, including normative associations of abstract parameters to SAML constructs specified using XPath [XPath], can be found in the eduGAIN profile detailed specification [EGDPS].

3.2 Publishing and Retrieving Metadata

The MDS deals with the metadata information model as defined in the SAML 2.0 Metadata specification [SAMLMD]. To lookup, search for, and publish metadata eduGAIN uses the REST architectural model based on HTTP exchanges. REST fits the MDS model well and is simple. It also has the benefit of being compatible with other systems using HTTP to retrieve metadata from a location stored in DNS [SAMLMD], and at the same time adding support for sophisticated searching and publishing.

Every federation participating in eduGAIN MUST publish via the MDS metadata related to its local interfaces, such as identity providers, attribute authorities, and service providers. These interfaces are generally subordinated to a BE and therefore the published metadata concerns one or more BEs from the local federation, with their associated interface descriptions. Typical metadata MUST include the component identifiers and location (contact URLs) of the corresponding elements, and SHOULD include additional information such as attributes supported or required by the specified interfaces.

This information is published in the form of SAML 2.0 XML documents having as root either an `EntityDescriptor` element (associated with one BE) or an `EntitiesDescriptor` element (associated to several BEs). Publication can take place from a centralised point within a federation (the FPP) or from BEs that have been authorised to publish their own metadata at the MDS. In the latter case, the root of the SAML 2.0 document is an `EntitiesDescriptor` containing data about one or more BEs from the local federation, while in the first case the root is an `EntityDescriptor` carrying the metadata associated to the publishing BE.

All published metadata are stored in a database at the MDS and can be retrieved upon inquiry from remote BEs during the trust establishment process. An interrogating BE may need for instance to locate the identity provider associated to a user from a different federation and therefore issues a metadata query based on certain information obtained from the user. The MDS response will consist of a SAML 2.0 document containing the required information about the user's home institution, or of an error message in case the search did not return a useful result.

A detailed description of the MDS REST profile, with the corresponding HTTP operations and return codes can be found in the eduGAIN profile specification [EGDPS].

3.3 Web Single Sign On (SSO) Profile

This profile is intended to cover all the use cases in which a human user, by means of a Web browser, pretends to access different eduGAIN-enabled resources, employing a single authentication appropriately fulfilled at the corresponding local identity provider.

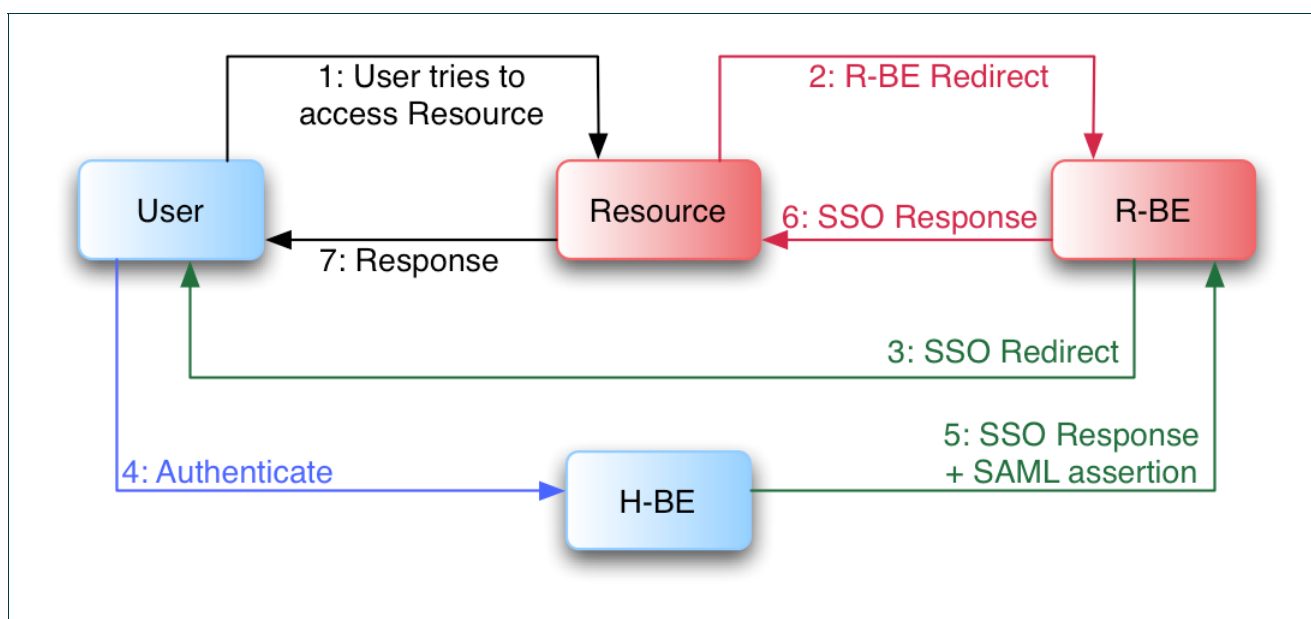


Figure 3.2: Message flow in the WebSSO profile

For those eduGAIN BEs configured to use SAML 1.1, Web SSO procedures MUST comply with those described by the Shibboleth Web SSO Browser/POST profile (as described in [SAMLBind] and [ShibArch]). When configured to use SAML 2.0, eduGAIN BEs acting in WebSSO scenarios SHALL apply the procedures described by the SSO profiles of SAML 2.0, as described by [SAML2Prof] and [SAML2Bind]

In both cases, the specific rules described in the profile detailed specification [EGDPS] SHALL be applied.

3.4 Automated Client Profile

This profile is indicated for the cases of software not directly operated by humans in the moment when it has to engage in an authentication or authorization interaction. This category includes elements such as daemons, autonomous servers, programs subject to automatically scheduled execution, etc.

Each automated client MUST have an X.509 certificate issued by a CA subordinated to one of the eduGAIN roots of trust. The client MUST send along with its request a proof that it is the legitimate owner of this certificate using the WS-Security X509 Certificate Token Profile [WSCTP], according to the processing rules described in the profile detailed specification [EGDPS].

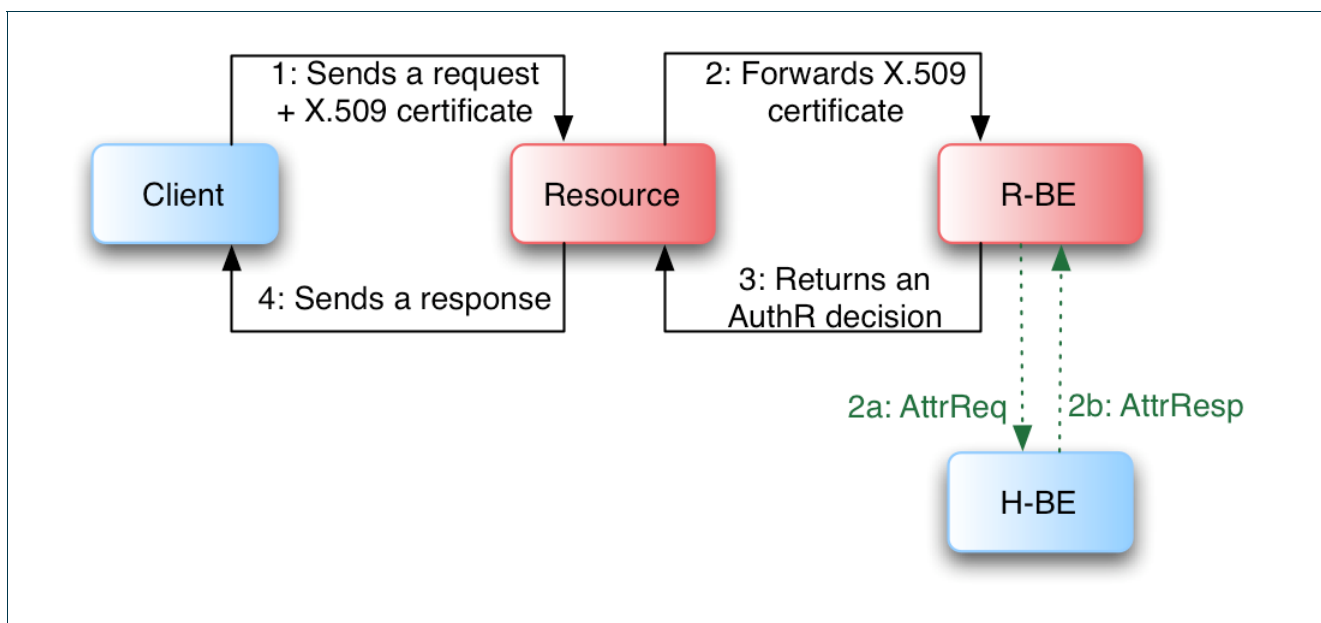


Figure 3.3: Message flow in the AC profile

3.5 User behind a Client Profile

This profile is applicable to Web Services clients that run independently of an HTTP (Web or application) server, but under the direct control of a human. These clients are standalone programs that can be freely installed at individual workstations or shared computers and are operated under direct control of their users.

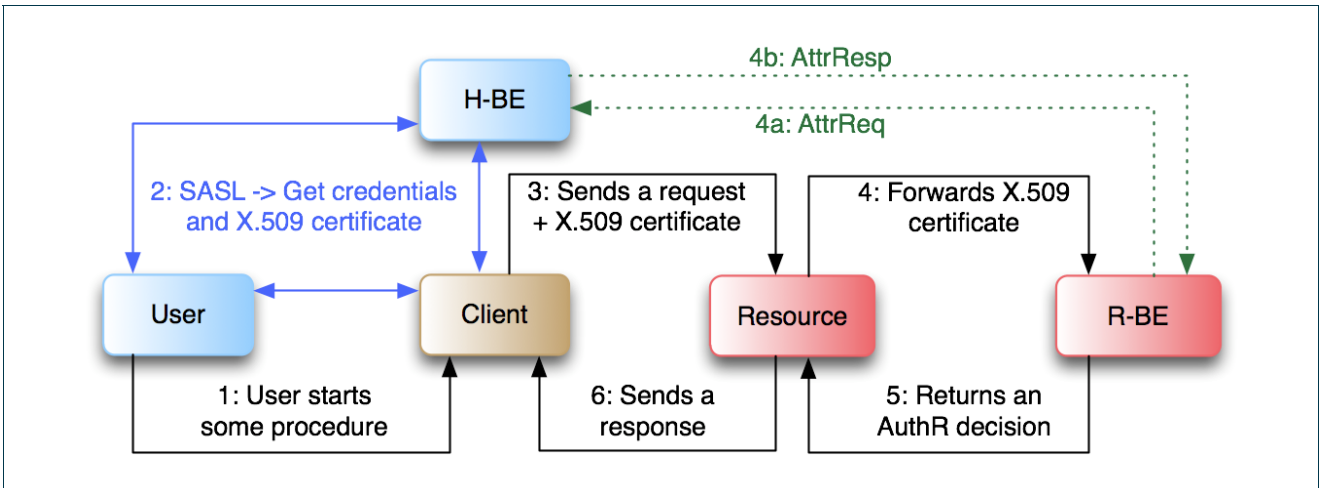


Figure 3.4: Message flow in the UbC profile

The processing rules applicable to this profile are the same described in the section above, the only difference being the way in which the client obtains the certificate used to identify the requester entity. In this case, the client must be able to act in the name of the individual using it in the moment the request is issued.

The current version of the profile mandates the client to use the SASL protocol [SASL] to send user credentials and get a temporary X.509 certificate. A more elaborated version, using new results of the GN2 JRA5 activity is under development.

3.6 Client in Web Container Profile

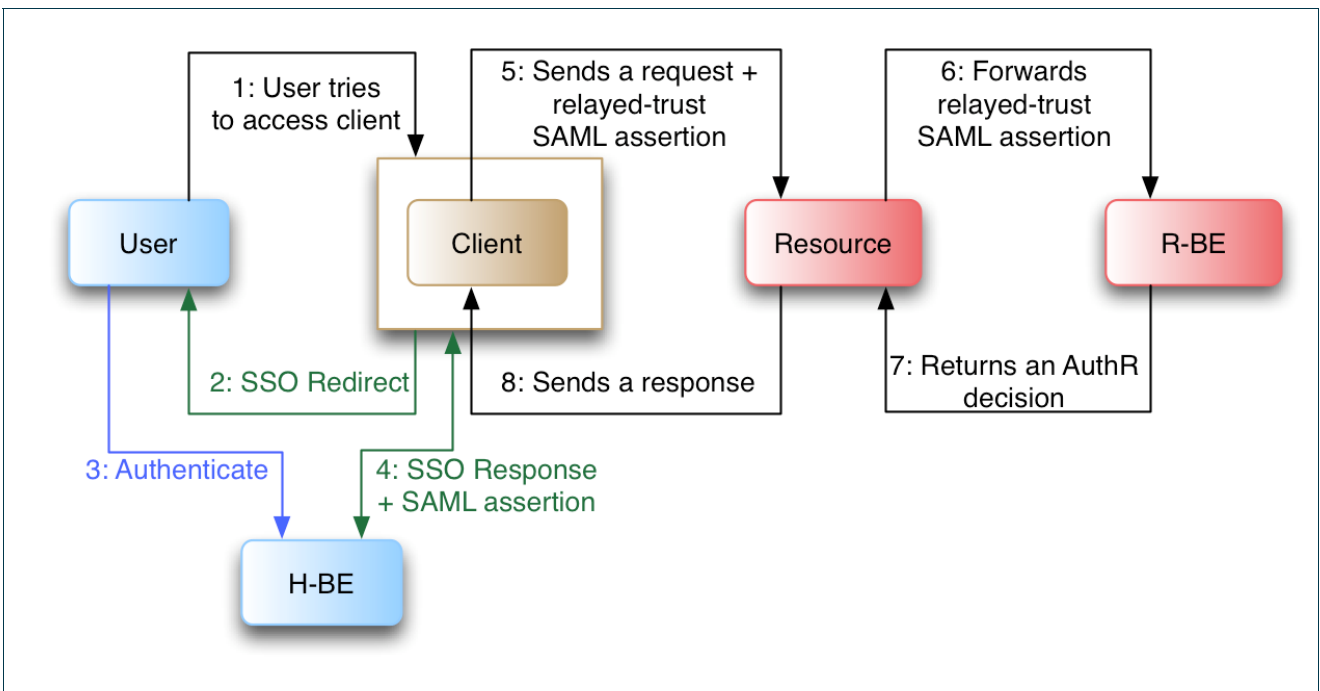


Figure 3.5: Message flow in the WE profile

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

This profile (for which the acronym *WE*, standing for “Web Enabled” will be used) is applicable in those cases where a certain software component (the client) is accessed by end users through a Web container (an application server, for example), and the client acts on behalf of the end user when requesting services to other component(s). To access the client, users must pass through the procedures described in the WebSSO profile, so the container can provide the client with the attributes received during the WebSSO phase.

In this case, user authentication is performed by means of the same Web browser used to access the client. After the WebSSO steps, the client is able to send a proof of user’s identity, as asserted by the H-BE and with the appropriate restrictions to avoid abuse. In summary, the profile provides a method for performing secure identity delegation through WebSSO, described in full detail in [EGDPS].

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

4 The eduGAIN Trust Fabric

A trust model is required in order to allow each eduGAIN component to assess the identity of its peer(s) during any interaction. This section describes this trust model, including the validation strategies for connections and signatures that must be followed. The trust establishment process will be enabled by means of using TLS connections for each eduGAIN interaction and including XML-Sig digital signatures for the appropriate protocol elements and assertions.

eduGAIN inter-component trust will be supported by a Public Key Infrastructure (PKI) based on X.509 certificates. It will be rooted at a set of Certification Authorities (CA) created and maintained within the project. This set will be referred to as the **eduGAIN truststore** and all eduGAIN components SHALL accept any of the CAs contained by the truststore as valid roots of trust. CAs in the eduGAIN truststore MUST conform to the **eduGAIN Certificate Policy**, a document defining the rules and procedures agreed by the eduGAIN participants to rely on digital public certificates issued to the components of the infrastructure.

At least one of these CAs will be specifically established and run by the project. This root CA will be referred as the **eduGAINCA**. The self-signed certificate of the eduGAINCA SHALL be the minimum content of the eduGAIN truststore.

4.1 PKI Structure

The structure of the eduGAIN PKI is shown in Figure 4.1. Each CA inside the eduGAIN truststore (shown as “Acc CA” in the figure) SHALL be accredited to issue certificates for components in a particular branch of the eduGAIN component identifier namespace (shown as “CId” in the figure). Certificates for components outside these branches SHALL be under the eduGAIN CA. The eduGAINCA SHALL only issue certificates to other CAs, and these subordinated CAs will in turn be responsible for issuing certificates to the individual components. The eduGAIN infrastructure SHALL provide at least one of these subordinated CAs, known as the **eduGAINSCA**.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

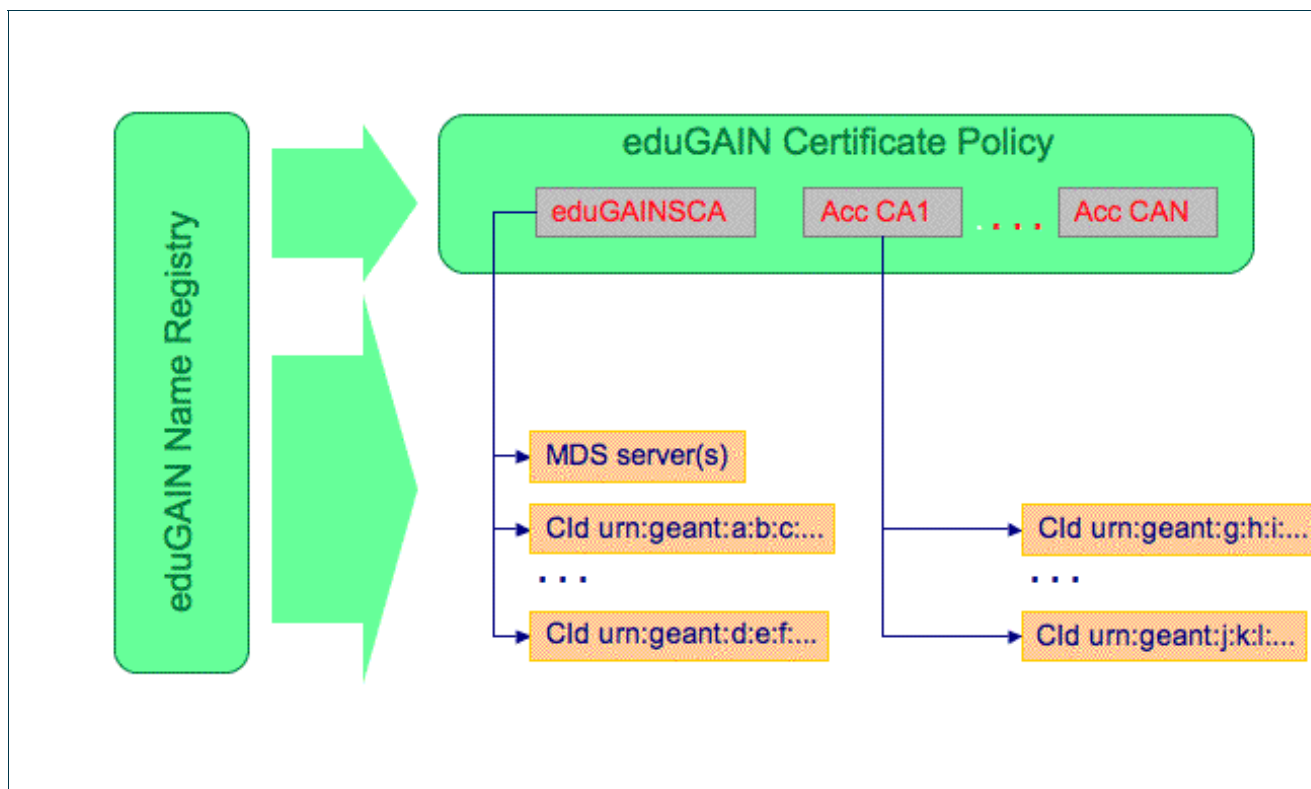


Figure 4.1: Structure of the eduGAIN PKI

The eduGAINSCA is able to provide a set of separately managed Registration Authorities (RA), according to the management procedures of the different eduGAIN namespaces under its responsibility.

4.2 Trust Validation Procedures

Trust validation MUST be performed by eduGAIN components according to a two-step procedure:

1. The received certificate SHALL be evaluated to check whether the whole trust path correctly resolves to the eduGAIN root of trust.
2. The eduGAIN component identifier contained in the Subject Alternate Name extension of the received certificate SHALL be evaluated against the metadata available for this interaction. It MUST match with the component identifier as stored in these metadata.

A failure in any of the verifications above SHALL cause a rejection of the requested operation with a `TrustError` result.

This procedure implies that, for a proper trust evaluation, all metadata exchange through the MDS MUST contain the eduGAIN component identifiers applicable in each case.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

Unless otherwise specified in the corresponding profile, all connections between any two eduGAIN components MUST use TLS and perform two-way certificate validation (both initiator and responder) according to the above procedures.

XML Signatures MUST be used in the following SAML constructs:

- Assertions containing one SAML `AuthenticationStatement` and (optionally) several SAML `AttributeStatement` in response to an eduGAIN `AuthenticationRequest`.

XML Signatures SHOULD be used in the following SAML constructs:

- Assertions containing SAML `AttributeStatement` in response to an eduGAIN `AttributeRequest`.

Validation of the certificates associated with XML Signatures MUST follow the procedures described above.

5 Software for Connecting to eduGAIN

At the moment of this writing at least three different pieces of software can be used to deploy an eduGAIN BE able to support at least one of the profiles described in the previous sections. These are:

- The eduGAIN API, built by the team building the whole confederation architecture as a reference implementation. It provides support for all the profiles discussed in this document.
- simpleSAMLphp [simpleSAML], that provides support to SAML-based identity infrastructures according to principles of simple deployment and extension. It provides support for the two mappings of the WebSSO profile
- Shibboleth 2 [Shib2], the SAML-based identity suite developed by a community around Internet2 and the most widely used by the international academic community. It can potentially provide support for both mappings of the WebSSO profile.

5.1 The eduGAIN API

The current implementation of the eduGAIN API is made in Java, and it provides a set of common libraries for all eduGAIN components. The eduGAIN API structure follows a layered approach, as shown below.

The eduGAIN API provides a general abstraction layer for authentication and authorisation operations, that is usable both by components directly woven into the eduGAIN trust fabric and by other elements within their internal trust domains, inside the participating federations or even locally.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

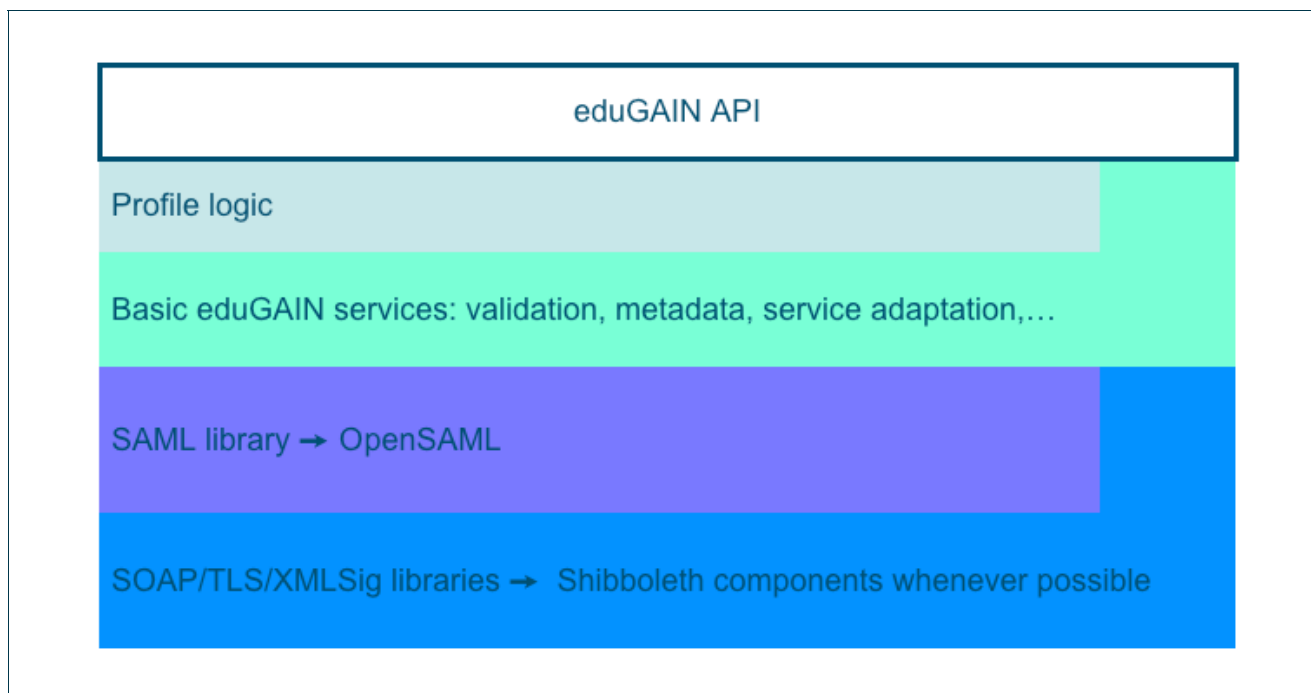


Figure 5.1: Layer diagram for the eduGAIN API

The basic layer includes all the basic support libraries for the different protocols that eduGAIN uses, as required by the current profiles and the binding libraries.

The layer above provides basic binding services for the protocols that implement the eduGAIN service definition. As the currently available binding is based on SAML, the eduGAIN API makes use of [OpenSAML].

The basic eduGAIN services are grouped into three Java packages:

1. **eduGAINVal**, providing the services required to carry out the eduGAIN trust validation procedures, and to prepare the exchanged material according to those procedures.
2. **eduGAINMeta**, providing the services required to publish and retrieve eduGAIN metadata.
3. **eduGAINBase**, providing the implementation of the eduGAIN abstract service definition plus specific interfaces for each of the defined profiles.

Profile logic is provided by specific packages inside eduGAINBase, including the *vanilla* eduGAIN profile corresponding to SAML over SOAP over HTTPS, according to the mapping defined by the general eduGAIN architecture specification.

A normal eduGAIN component should only need to interface with eduGAINBase, though any of the above listed services can be accessed if required. The interface is based on the use of specific classes modelling the abstract interface described in [GN2DJ522] and provided by the eduGAINBase package.

6 A Checklist for Connecting to eduGAIN

Although eduGAIN has not reached the level of production service yet, it is currently a stable infrastructure connecting more than ten European national identity federations, providing service to several distributed multi-domain applications, and running test interconnections with other federated identity services worldwide. Along this section a set of steps to connect a set of resources and/or identity repositories to the eduGAIN confederated infrastructure is detailed. It is not intended to be an exhaustive user guide, but to provide a comprehensive list of the steps that must be accomplished:

- Establish an identity federation. A federating software must be selected (either free or commercial) and an internal policy and procedures established. eduGAIN is federation agnostic, as long as the appropriate BEs are in place for a given federating solution and procedures do not break the eduGAIN confederation policy. BEs produced along the development of eduGAIN are available for A-Select, PAPI, Shibboleth and the Sun Federation Suite, although many others may exist. For more help, refer to the [REFEDS] group.
- Request the appropriate identifiers from the eduGAIN Naming Registry, or request the delegation of a branch in the eduGAIN namespace. All component identifiers of the elements connecting the federation to eduGAIN must be derived from this identifier according to the guidelines in section 3. The eduGAIN Naming Registry is available at <http://registry.edugain.org/>
- Entangle the federation in the eduGAIN trust fabric. This can be done either by accrediting the federation CA to be included in the eduGAIN truststore, or by applying for certificates through eduGAINSCA. In a first stage, it is recommended to start with the latter and consider the possibility of accrediting a federation CA later on. Data on eduGAIN PKI is available at <http://pki.edugain.org/> and the eduGAINSCA site can be found at <http://sca.edugain.org/>
- Design the structure of the eduGAIN connection. It is recommended to start with a highly centralised schema, with a single FPP and a single BE acting as Local Federation Adaptor as the unique points of contact between the federation and eduGAIN. With more experience a more distributed and flexible setup can be created.
- Validate the eduGAIN connection by means of the eduGAIN Validation Facility. This is a set of testing components that help tune the components to what eduGAIN expects from them. It is mandatory to pass the validation tests before a federation can connect to the production eduGAIN infrastructure. More on this can be found at the eduGAIN site, <http://www.edugain.org/>
- Collaborate with the rest of eduGAIN participants in maintaining and enhancing the infrastructure. eduGAIN is a collaborative project that needs support. Community resources are available at the eduGAIN wiki [JRA5WIKI].

7 Conclusions

The “eduGAIN cookbook” provides the essential information about eduGAIN technological procedures and requirements that needs to be understood and applied by the federations wishing to participate. The core concepts forming the base of the eduGAIN architecture are introduced, together with a more detailed presentation of its components and their corresponding roles. The interactions between these components in order to ensure safe and trustworthy communication between local federations are explained within the framework of particular use cases, forming the eduGAIN profiles and protocols.

Special attention needs to be directed towards the eduGAIN trust fabric, as its correct implementation by all participants is an essential condition in the good functioning of the confederation infrastructure. Together with this information, the document aims to cover the necessary steps to be followed by local federations in order to successfully participate in eduGAIN.

Project:	GN2
Deliverable Number:	DJ5.2.3,3
Date of Issue:	21/05/08
EC Contract No.:	511082
Document Code:	GN2-08-130

8 References

- [GN2DJ511]** JRA5 Glossary of terms
<http://intranet.geant2.net/server/show/conMediaFile.6254>
- [GN2DJ522]** D. Lopez, R. Castro, B. Kerver, T. Lenggenhager, I. Melve, M. Milinovic, J. Rauschenbach, K. Wierenga, S. Winter, H. Ziemek et al. GÉANT2 Authentication and Authorisation Infrastructure (AAI) Architecture. GÉANT2 Deliverable DJ5.2.2. October 2005.
http://www.geant2.net/upload/pdf/GN2-07-024-DJ5-2-2-2-GEANT2_AAI_Architecture_And_Design.pdf
- [JRA5Wiki]** Collaboration site for the GÉANT2 JRA5 participants, wiki.rediris.es/jra5
- [OpenSAML]** OpenSAML 1.1/2.0 - an Open Source Security Assertion Markup Language implementation.
<http://www.opensaml.org/>
- [REFEDS]** REFEDS: Research and Education Federations. Group under the auspices of the TERENA Technical Programme on Middleware. <http://www.terena.nl/activities/refeds/>
- [REST]** Representational State Transfer,
http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
- [RFC2119]** S. Bradner. Key words for use in RFCs to Indicate Requirement Levels. Internet Best Current Practice, IETF. March 1997.
- [RFC3280]** R. Housley et al., Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF, April 2002.
- [SAML11]** E. Maler et al. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V1.1. OASIS Standard, September 2003.
<http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf>
- [SAML20]** S. Cantor et al. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, March 2005.
<http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- [SAML2Bind]** Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, March 2005.
<http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf>

- [SAML2Prof]** Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, March 2005.
<http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf>
- [SAMLBind]** E. Maler et al. Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML). OASIS Standard, September 2003.
<http://www.oasis-open.org/committees/download.php/3405/oasis-sstc-saml-bindings-profiles-1.1.pdf>
- [SAMLMD]** S. Cantor (editor). Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, March 2005.
<http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf>
- [SASL]** RFC2222. Simple Authentication and Security Layer (SASL)
- [simpleSAML]** Feide RnD. simpleSAMLphp <http://rnd.feide.no/simplesamlphp>
- [SCHAC]** J. Masa (editor). SCHAC Attribute Definitions for Individual Data, May 2006.
<http://www.terena.org/activities/tf-emc2/docs/schac/schac-schema-IAD-1.3.0.pdf>
- [Shib2]** Internet2 Middleware. Shibboleth Project. <http://shibboleth.internet2.edu/>
- [ShibArch]** S. Cantor (editor). Shibboleth Architecture. Protocols and Profiles.10 September 2005.
<http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-arch-protocols-200509.pdf>
- [WSCTP]** Web Services Security X.509 Certificate Token Profile. OASIS Standard, March 2004.
<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0.pdf>
- [XPATH]** J. Clark, S. DeRose (editors), XML Path Language (XPath) Version 1.0. W3C Recommendation, November 1999.
<http://www.w3.org/TR/xpath>

9 Acronyms

In JRA5 used acronyms can be found in the JRA5 Glossary of Terms [DJ5.1.1]. Often used terms are listed below.

AA	Attribute Authority
AAI	Authentication and Authorisation Infrastructure
API	Application Programming Interface
BE	Bridging Element
CA	Certificate Authority
DN	Distinguished Name
FPP	Federation Peering Point
FQDN	Fully Qualified Domain Name
IdP	Identity Provider
LA	Local Adapter
LFA	Local Federation Adapter
MDS	Metadata Service
PKI	Public Key Infrastructure
REST	Representational State Transfer
SAML	Security Assertion Mark-up Language
SASL	Simple Authentication and Security Layer
SCHAC	SChema Harmonisation Committee
SOAP	Simple Object Access Protocol
SP	Service Provider
TLS	Transport Layer Security
URL	Unified Resource Locator
URN	Unified Resource Names
XML	eXtensible Mark-up Language