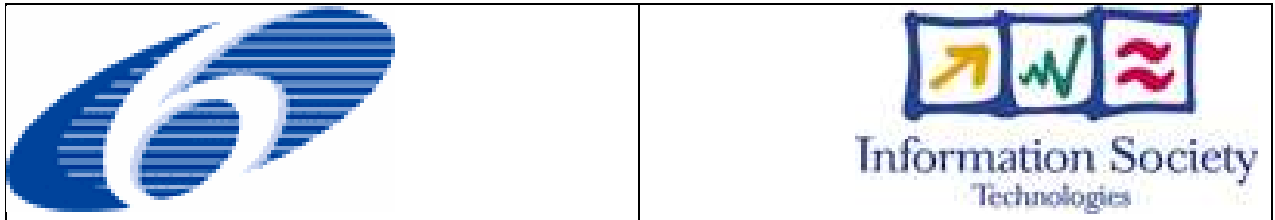


**Sixth Framework Programme
Priority IST 2.5.12
Information Society Technologies**



Integrated Project



Contract No.: 033564

**Specification of the Identity Management and Authentication
Service**

Version 1.1

Due date of deliverable: 31/05/2009

Internal release date: 24/03/2008
Actual submission date: 24/03/2008

Document Control Page

Title	Specification of the Identity Management and Authentication Service	
Creator	EIG	
Editor	Pascal Dihé (EIG)	
Description	This document defines an implementation specification of the Identity Management and Authentication Service for the SANY Web Services Platform.	
Publisher	SANY Consortium	
Contributors	Julian Fischer (EIG) Thomas Berlinghoff (EIG) Thorsten Herter (EIG) Nils Steinbiß (EIG) Wenjie Ma (EIG)	
Type	Text	
Format	MS Word	
Language	EN-GB	
Creation date	2006-08-10	
Version number	1.1	
Version date	2008-02-02	
Last modified by	EIG	
Rights	Copyright "SANY Consortium". During the drafting process, access is generally limited to the SANY Partners.	
Audience	<input type="checkbox"/> internal <input checked="" type="checkbox"/> public <input type="checkbox"/> restricted, access granted to:	
Review status	<input type="checkbox"/> Draft <input checked="" type="checkbox"/> WP Manager accepted <input type="checkbox"/> SP Manager accepted <input type="checkbox"/> MB quality controlled <input type="checkbox"/> Co-ordinator accepted	Where applicable: <input type="checkbox"/> Accepted by the GA <input type="checkbox"/> Accepted by the GA as public document
Action requested	<input type="checkbox"/> to be revised by Partners involved in the preparation of the Project Deliverable <input type="checkbox"/> to be revised by all SANY Partners <input type="checkbox"/> for approval of the WP Manager <input checked="" type="checkbox"/> for approval of the SP Manager <input type="checkbox"/> for approval of the Quality Manager <input type="checkbox"/> for approval of the Project Co-ordinator <input type="checkbox"/> for approval of the General Assembly	
Requested deadline	<<dd/mm/yyyy >>	

Copyright © 2009, SANY Consortium

The SANY Consortium (www.sany-ip.eu) grants third parties the right to use and distribute all or parts of this document, provided that the SANY project and the document are properly referenced.

THIS DOCUMENT IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Table of Contents

1. Introduction	10
2. Overview and Architecture Outline.....	11
2.1. Role and Scope of the Identity Management and Authentication Service	11
2.1.1 Identities and Session Information	11
2.2. Service Specification Summary.....	12
3. Context of the Identity Management and Authentication Service.....	13
3.1. Relations to Standards	13
3.2. Relations to Information Models.....	13
3.3. Relations to other Service Specifications	13
4. Specification of the Service Capabilities Interface.....	14
4.1. getCapabilities Operation	14
4.1.1 OA_GetCapabilitiesRequest Type	16
4.1.2 OA_GetCapabilitiesResponse Type.....	16
5. Specification of the Authentication Interface	18
5.1. login Operation	18
5.1.1 SAML AuthnRequest Type.....	19
5.1.2 SAML Response Type	20
5.2. verifySessionInformation Operation	22
5.2.1 verifySessionInformationRequest Type.....	23
5.3. verifySessionInformationResponse Type	23
6. Specification of the Identity Management Interface.....	25
6.1. activateIdentity Operation.....	25
6.1.1 activateIdentityRequest Type	26
6.2. deactivateIdentity Operation.....	27
6.2.1 deactivateIdentityRequest Type	28
6.3. createIdentity Operation	29
6.3.1 createIdentityRequest Type	30
6.4. deleteIdentity Operation	33
6.4.1 deleteIdentityRequest Type.....	34
6.5. updateIdentity Operation	35
6.5.1 updateIdentityRequest Type	36
6.6. addCredentials Operation	37
6.6.1 addCredentialsRequest Type.....	38
6.7. updateCredentials Operation	40
6.7.1 updateCredentialsRequest Type.....	41
6.8. deleteCredentials Operation.....	42
6.8.1 deleteCredentialsRequest Type	43
6.9. getIdentitiesOperation	44
6.9.1 getIdentitiesRequest Type	45
6.9.2 getIdentitiesResponse Type.....	45

7. References	47
8. Appendix A: XML Schema and WSDL Documents	49
8.1. XML Schema Documents.....	49
8.1.1 identity_authen_exceptions.xsd	49
8.1.2 identity_authen__types.xsd.....	51
8.1.3 identity_authen_requests.xsd.....	56
8.2. WSDL Document.....	59
8.3. Capabilities Document Template	70

Table of Acronyms

LDAP	Lightweight Directory Access Protocol
OASIS	1) Open Advanced System for Disaster and Emergency Management 2) Organization for the Advancement of Structured Information Standards
OGC	Open Geospatial Consortium
ORCHESTRA	Open Architecture and Spatial Data Infrastructure for Risk Management
OSI	ORCHESTRA Service Instance
PDP	Policy Decision Point
PEP	Policy Enforcement Point
RBAC	Role Based Access Control
SAML	Security Assertion Markup Language
SensorSA	Sensor Service Architecture
UAA	User Management, Authentication and Authorisation
WSDL	Web Services Description Language
WSS	Web Services Security
XACML	eXtensible Access Control Markup Language
XML	eXtensible Markup Language

Tables

Table 1: Specification of the getCapabilities Operation	15
Table 2: Specification of the login Operation	18
Table 3: Specification of the verifySessionInformation Operation	22
Table 4: Specification of the activateIdentity Operation	26
Table 5: Specification of the deactivateIdentity Operation.....	27
Table 6: Specification of the createIdentity Operation	29
Table 7: Specification of the deleteIdentity Operation	34
Table 8: Specification of the updateIdentity Operation	35
Table 9: Specification of the addCredentials Operation	38
Table 10: Specification of the updateCredentials Operation	40
Table 11: Specification of the deleteCredentials Operation.....	42
Table 12: Specification of the getIdentitiesOperation.....	44

Diagrams

Diagram 1: Simplified Class Diagram of the Identity Management and Authentication Service 12

Figures

Figure 1: getCapabilities Operation	15
Figure 2: OA_GetCapabilitiesRequest.....	16
Figure 3: OA_GetCapabilitiesResponse	17
Figure 4: login operation	19
Figure 5: SAML AuthnRequest	19
Figure 6: SAML Subject Type	20
Figure 6: SAML Response.....	21
Figure 7: verifySessionInformation operation	22
Figure 8: verifySessionInformationRequest.....	23
Figure 9: verifySessionInformationResponse	24
Figure 10: activateIdentity operation	26
Figure 11: activateIdentityRequest.....	26
Figure 12: deactivateIdentity operation.....	28
Figure 13: deactivateIdentityRequest.....	28
Figure 14: createIdentity operation	30
Figure 15: createIdentityRequest	30
Figure 16: UsernameIdentity Type	31
Figure 17: AttributedIdentity Type	32
Figure 18: KeyVectorIdentityAttributes Type	32
Figure 19: deleteIdentity operation	34
Figure 20: deleteIdentityRequest	34
Figure 21: updateIdentity operation	36
Figure 22: updateIdentityRequest	36
Figure 23: addCredentials operation	38
Figure 24: addCredentialsRequest	38
Figure 25: PasswordCredentials Type.....	39
Figure 26: updateCredentials operation	41
Figure 27: updateCredentialsRequest.....	41
Figure 28: deleteCredentials operation	43
Figure 29: deleteCredentialsRequest.....	43
Figure 30: getIdentities operation	45
Figure 31: getIdentitiesRequest.....	45
Figure 32: getIdentitiesResponse	46

1. Introduction

The present version 1.0.1 of the Identity Management and Authentication Service is an advancement of the former specification of the Authentication Service. It has been heavily reworked and consequently enhanced. The former specification of the Authentication Service was originally based on

- the ORCHESTRA abstract specification of the Authentication Service, Version 1.3 and
- the ORCHESTRA implementation specification of the Authentication Service, Version 1.1

Since the Authentication Service has originally been defined in the ORCHESTRA project as an integral part of the ORCHESTRA UAA concept, knowledge of this concept was essential for the understanding of the former specification. Although the new specification of the Identity Management and Authentication Service does not have this restriction, a basic understanding of the concept of profiles and identities is required. Therefore those concepts and also the collaboration of the SANY Access Control Services are explained briefly in the Specification of the Profile Management Service and more detailed in the Information Viewpoint of the SensorSA [SANY D2.3.3].

This document specifies the interfaces implemented by the Identity Management and Authentication Service as well as its main purpose and provides a formal description of the implemented operations in WSDL and XML-Schema in conformance to the guidelines and rules of the SANY W3C Web Services Platform defined in the SensorSA.

The changes to the original specification of the Authentication Service, the relations to SANY technical requirements and progress of the specification work with respect of the individual project phases are documented in the D2.4.x Deliverables.

2. Overview and Architecture Outline

2.1. Role and Scope of the Identity Management and Authentication Service

The Identity Management and Authentication Service is one of the four Access Control Services of the SensorSA. Together with the Profile Management Service and the underlying information model it forms an abstract user concept that allows the separation of user profiles and identities.

The Identity Management Service is responsible for the management of identities (create, update, delete) while the Authentication Interface is responsible for the verifications of genuineness of identities using a set of given credentials. The current specification of this service supports UsernameIdentities and PasswordCredententials and thus a username/password authentication scenario. As PasswordCredententials and UsernameIdentities are derived from generic base types the support for different authentication mechanisms is possible.

2.1.1 Identities and Session Information

Session information returned after a successful authentication can be used to invoke services demanding authenticated identities. A security-enabled service or a PEP (e.g. a Generic PEP Proxy) might use this information to perform authorisation requests against a PDP (e.g. an Authorisation Service). The session information issued by the present specification of the Identity Management and Authentication Service is encoded in SAML 2.0.

The Identity Management and Authentication Service is also able to verify a previously issued SAML Token. Each service that relies on session information should typically have an Authorisation Service, a security-enabled service, or a Generic PEP Service validate the session information before performing any security-related actions. In our context the transport of session information can be handled either as additional operation parameter or, when using the SOAP protocol, transparently in the SOAP header. The manner in which that session information is transported and how it is encrypted is beyond the scope of the specification of the Identity Management and Authentication Service and depends on the chosen platform. This issue is therefore addressed in the specification of the SANY W3C Web Services Platform and the Generic PEP Service.

Identities are managed in instances of the Identity Management and Authentication Service and are uniquely identified by an ID which also contains the URI of the Identity Management and Authentication Service instance. Multiple instances of Identity Management and Authentication Services may coexist in a network and each organisation may maintain their own installation of the Identity Management and Authentication Service. Cross-organisational or single-sign-on is easily supported since identities represent only the identity of a (user) profile and one profile may refer to multiple identities, each registered at different instances of the Identity Management and Authentication Service.

The Identity Management and Authentication Service supports heterogeneous security infrastructures with disparate authentication mechanisms. Different instances of Identity Management and Authentication Services may implement different authentication methods and may also define different types of identities.

2.2. Service Specification Summary

The implementation specification of the Identity Management and Authentication Service is comprised of the following abstract interfaces that are defined in distinct interface type specifications:

- The Service Capabilities Interface Type
- The Authentication Interface Type
- The Identity Management Interface Type

Operations related to the management of identities have been removed from the former ORCHESTRA Authentication Interface and are now specified in the new Identity Management Interface.

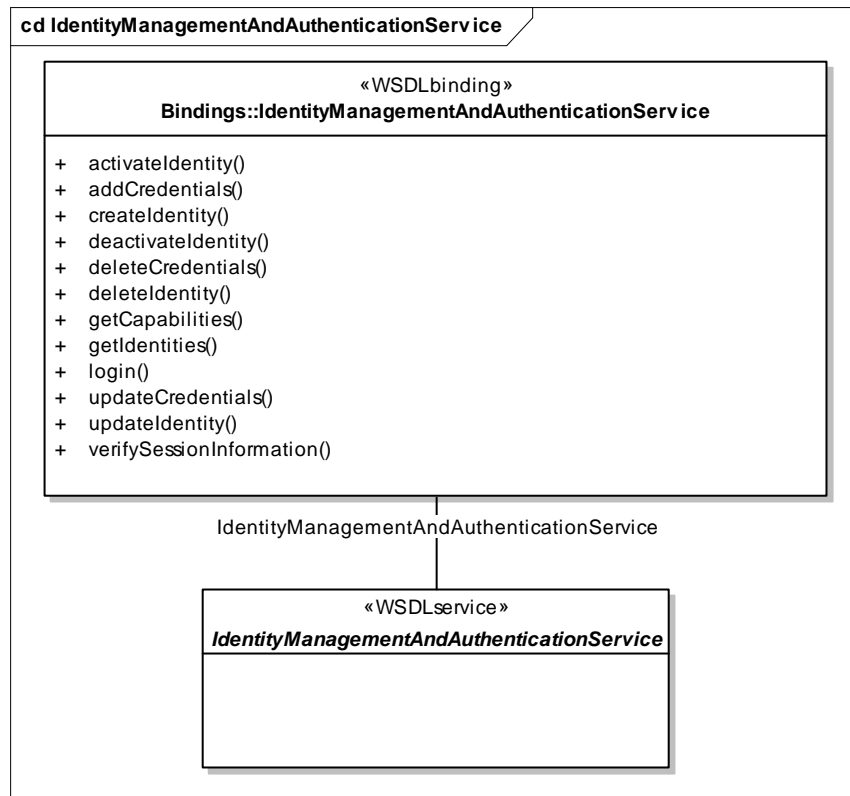


Diagram 1: Simplified Class Diagram of the Identity Management and Authentication Service

The formal platform-specific description of the Identity Management and Authentication Service can be found in Appendix A: XML Schema and WSDL Documents.

3. Context of the Identity Management and Authentication Service

3.1. Relations to Standards

Session Information is encoded in the SAML v2.0 OASIS Standard. There are numerous standards in this context that are relevant especially for the implementation:

- Java.sun.com Java Authentication and Authorization Service (JAAS) (part of Java 2 SDK 1.4). <http://java.sun.com/products/jaas/>
- OASIS Digital Signature Services (DSS) TC
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=dss
- OASIS Public Key Infrastructure (PKI) TC
http://www.oasis-open.org/committees/workgroup.php?wg_abbrev=pki
- OASIS Web Services Secure Exchange (WS-SX) TC
http://www.oasis-open.org/committees/workgroup.php?wg_abbrev=ws-sx
- OASIS Web Services Security (WSS) TC
http://www.oasis-open.org/committees/workgroup.php?wg_abbrev=wss
- Security Assertion Markup Language (SAML) V2.0
<http://wiki.oasis-open.org/security/Saml2TechOverview>

3.2. Relations to Information Models

Apart from the parameter types described in this document, there are no additional relations to information models.

3.3. Relations to other Service Specifications

The Identity Management and Authentication Service has a strong relation to the other SANY Access Control Services since its operations are per definition access controlled, except for the getCapabilities and the login operation.

4. Specification of the Service Capabilities Interface

The Service Capabilities Interface has originally been described in the ORCHESTRA Specification of the OA Basic Service. To maintain backward compatibility to ORCHESTRA Services (e.g. the Catalogue Service), this interface remains unchanged.

4.1. getCapabilities Operation

The specification of the getCapabilities operation has been copied without any modification from the respective ORCHESTRA interface specification. SANY specific changes are only required on the level of the capabilities document itself.

The mandatory getCapabilities operation informs the client of the capabilities of an service instance. This operation takes into account that in addition to capabilities that may be common to all services in a service network a service may provide a specific set of capabilities. Furthermore, this operation allows the capabilities to be delivered according to different service meta-information schemas. This implementation specification therefore does not prescribe a certain schema to be used for the service capabilities. One or several schemas to be supported as well as a default schema have to be defined in the context of a service network.

A service meta-information document is returned to the requesting client, either complete or including selected parts according to the given sections in the request.

A request to perform the getCapabilities operation shall include the parameters listed and defined in Table 1. This table also specifies the data type (Type), the obligation [optional | mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Identical to ORCHESTRA Specification			
Overrides	Not applicable			
Preconditions	none			
Post conditions	Service meta-information document returned to requesting client, either complete or including selected parts according to the given sections in the request			
Use	mandatory			
Receives	Name	Type	Use	Description
	request	OA_GetCapabilities Request	mandatory	Specifies the parts of the meta-information to be returned. If absent, all parts shall be returned using the default schema.

Returns	Type	Description
		OA_CapabilitiesDocument
Throws	Type	Cause
	OA_InvalidParameterValue	Operation request contains an invalid parameter value. Return the name of the parameter with invalid value.
	OA_MissingParameterValue	Operation request does not include a parameter value. Return the name of the missing parameter.
	OA_NoApplicableCode	No other basic or service-specific exception type applies.
	OA_InternalError	A problem occurred in the runtime environment (e.g. out of memory).
	OA_VersionNegotiationFailed	List of versions in acceptSpecVersions parameter value in the getCapabilities request did not include any version supported by the service instance.
	OA_UnsupportedSchema	The sections parameter in the getCapabilities request referred to a schema unsupported by the service instance.

Table 1: Specification of the getCapabilities Operation

The formal platform-specific specification of the getCapabilities operation can be found in Appendix A: XML Schema and WSDL Documents.

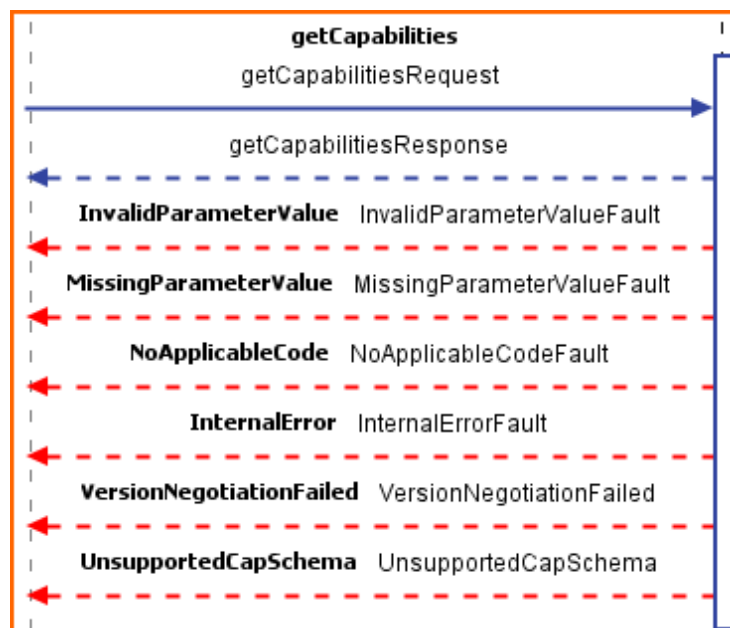


Figure 1: getCapabilities Operation

4.1.1 OA_GetCapabilitiesRequest Type

The OA_GetCapabilitiesRequest Type consists of the following elements:

- **acceptFormats:** Optional parameter containing a prioritized sequence of zero or more response formats desired by the caller, with preferred formats listed first. The formats are to be expressed in terms of MIME types. Independent of the contents of this element, the returned capabilities can always be formatted according to the default MIME type which is defined as “text/xml”.
- **acceptSpecVersions:** Optional parameter containing a prioritized sequence of zero or more specification versions of the service accepted by the client, with the preferred versions listed first. If no versions are included in the request, the highest version that the service supports shall be used.
- **sections:** Optional parameter specifying the schema for service meta-information according to which the capabilities shall be structured. In addition the names of requested sections in the complete set of meta-information elements can be listed. If absent, the complete service capabilities shall be returned structured according to a default schema.

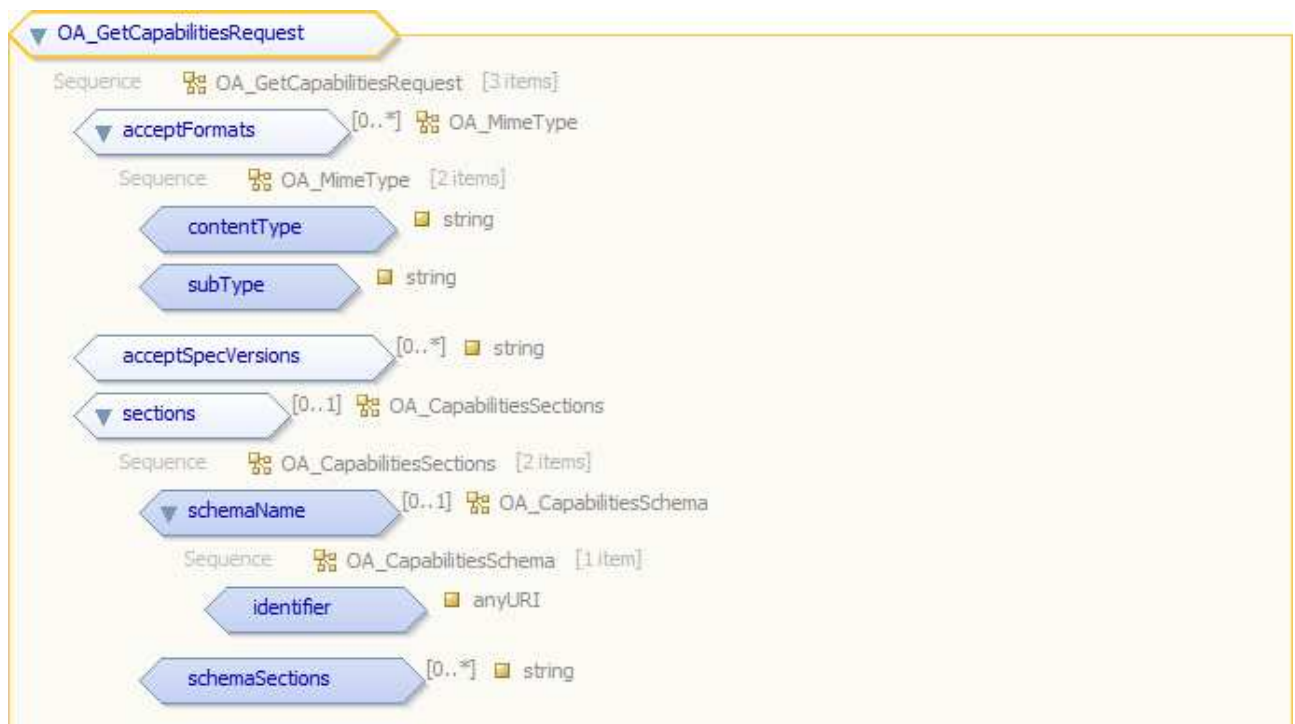


Figure 2: OA_GetCapabilitiesRequest

4.1.2 OA_GetCapabilitiesResponse Type

The OA_GetCapabilitiesRequest Type consists of the following elements:

- **capabilitySections:** Capabilities as meta-information document which is internally structured according to the indicated format and the indicated schema. It is either complete or includes only selected parts according to the given sections in the request.
- **format:** MIME type of the format in which the capabilities are returned as value of the capabilitySections parameter. The value of this attribute may always denote the default MIME Type “text/xml” indicating that the capabilities are formatted in XML.
- **schemaName:** Schema according to which the capabilities are returned. The value is the same as the one contained in the sections parameter of the request. If not explicitly included in the request, the value indicates the default schema.
- **version:** Version number of the specification to which the delivered service meta-information document is conform.

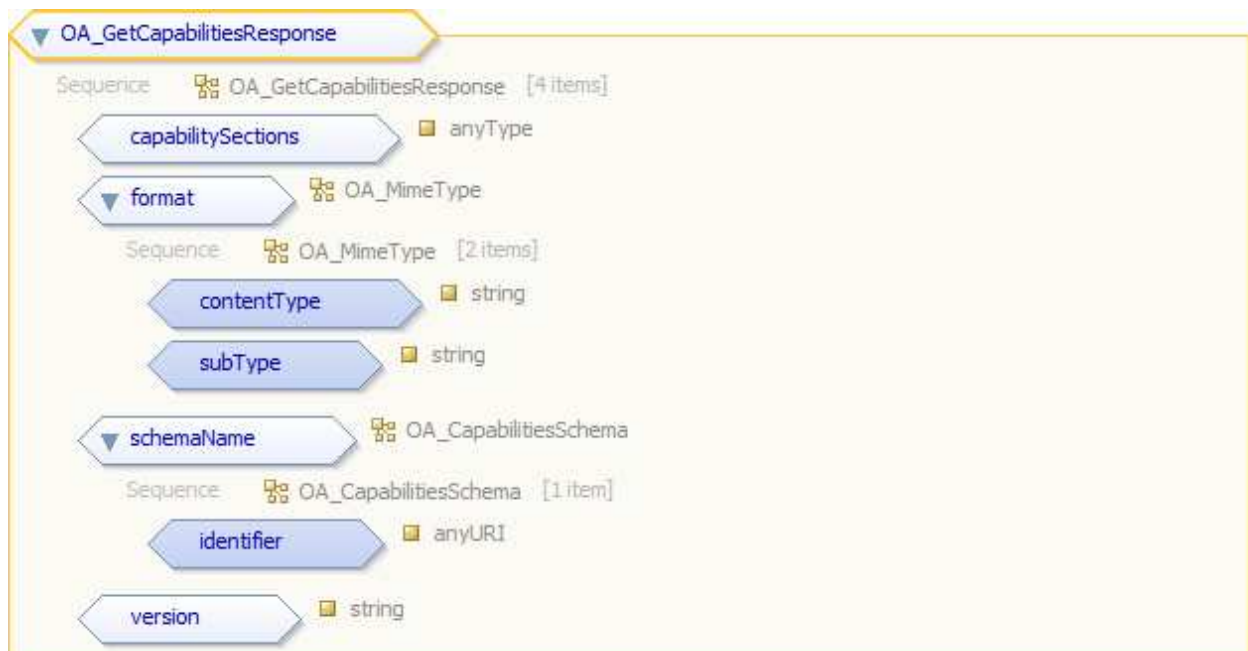


Figure 3: OA_GetCapabilitiesResponse

A template for the service specific response of the getCapabilities operation is specified in chapter 8.3.

5. Specification of the Authentication Interface

The Authentication Interface defines operations to prove the genuineness of identities using a set of given credentials and to issue session information after successful login.

5.1. login Operation

The mandatory login operation initiates the validation of a certain identity for given credential. It returns session information in form of a SAML 2.0 ticket which contains assertions for the authenticated identity as well as assertions for the group identities to which this identity is associated.

A request to perform the login operation shall include the parameters listed and defined in Table 2. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	None			
Post conditions	None			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	AuthnRequest	mandatory	The SAML 2.0 AuthnRequestType.
Returns	Type		Description	
	ResponseType		A SAML 2.0 response, which contains the authenticated SAML Ticket.	
Throws	Type		Cause	
	none		none	

Table 2: Specification of the login Operation

The formal platform-specific specification of the login operation can be found in Appendix A: XML Schema and WSDL Documents.

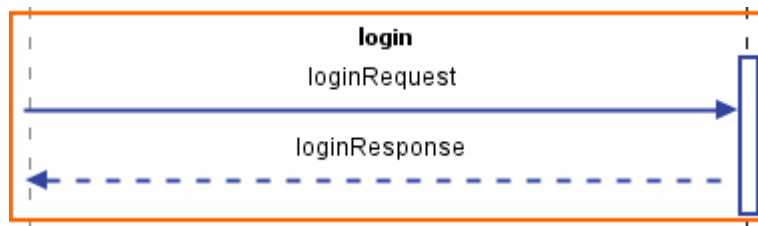


Figure 4: login operation

5.1.1 SAML AuthnRequest Type

The loginRequest is an instance of the AuthnRequest element which is defined in [OASIS-SAML] as follows:

“To request that an identity provider issues an assertion with an authentication statement, a presenter authenticates to that identity provider (or relies on an existing security context) and sends it an <AuthnRequest> message that describes the properties that the resulting assertion needs to have to satisfy its purpose. Among these properties may be information that relates to the content of the assertion and/or information that relates to how the resulting <Response> message should be delivered to the requester. The process of authentication of the presenter may take place before, during, or after the initial delivery of the <AuthnRequest> message.

The requester might not be the same as the presenter of the request if, for example, the requester is a relying party that intends to use the resulting assertion to authenticate or authorize the requested subject so that the relying party can decide whether to provide a service.”



Figure 5: SAML AuthnRequest

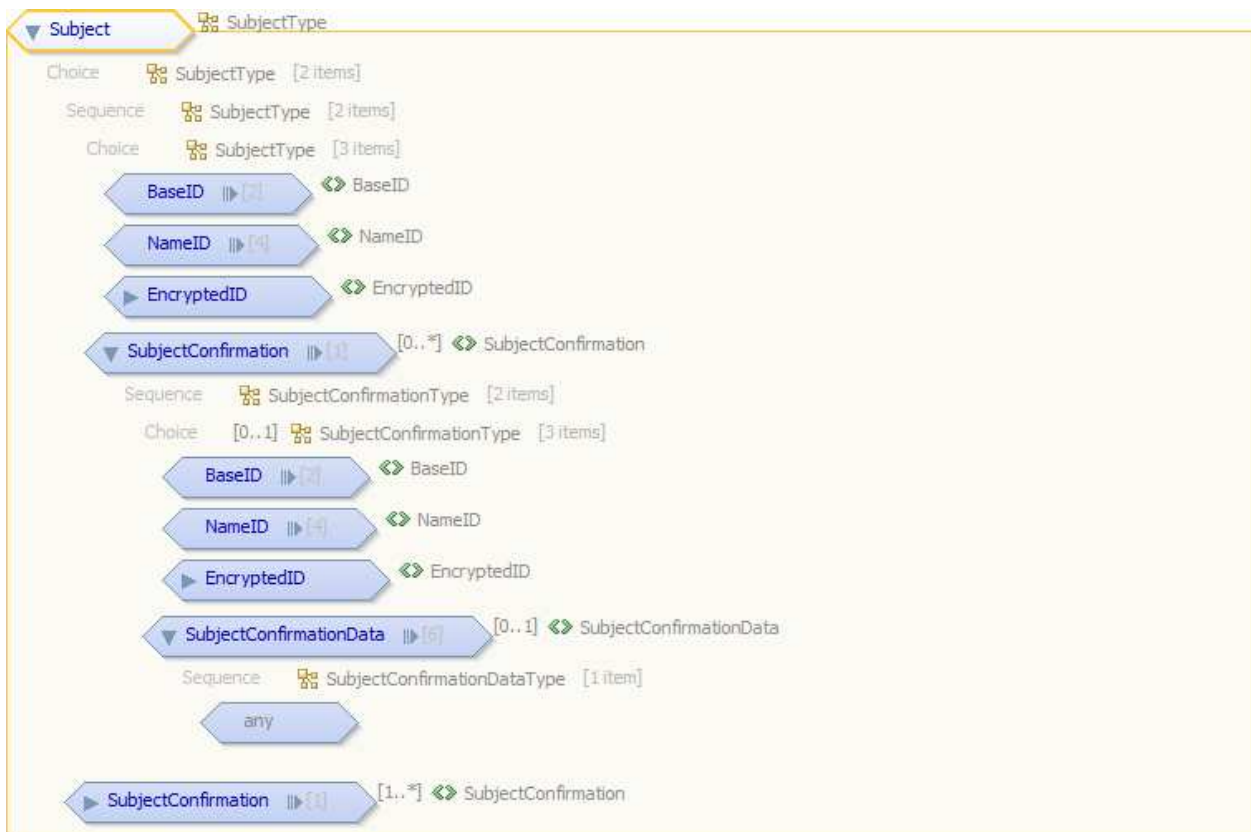


Figure 6: SAML Subject Type

An identity corresponds to the SAML Subject Type and the credentials correspond to the SubjectConfirmationData.

5.1.2 SAML Response Type

The loginResponse is an instance of the Response element which is defined in [OASIS-SAML].

- Status: Specifies whether the authentication could successfully be performed, or not.
- Assertion: The sessionInformation (SAML Ticket) as issued through the authentication process.

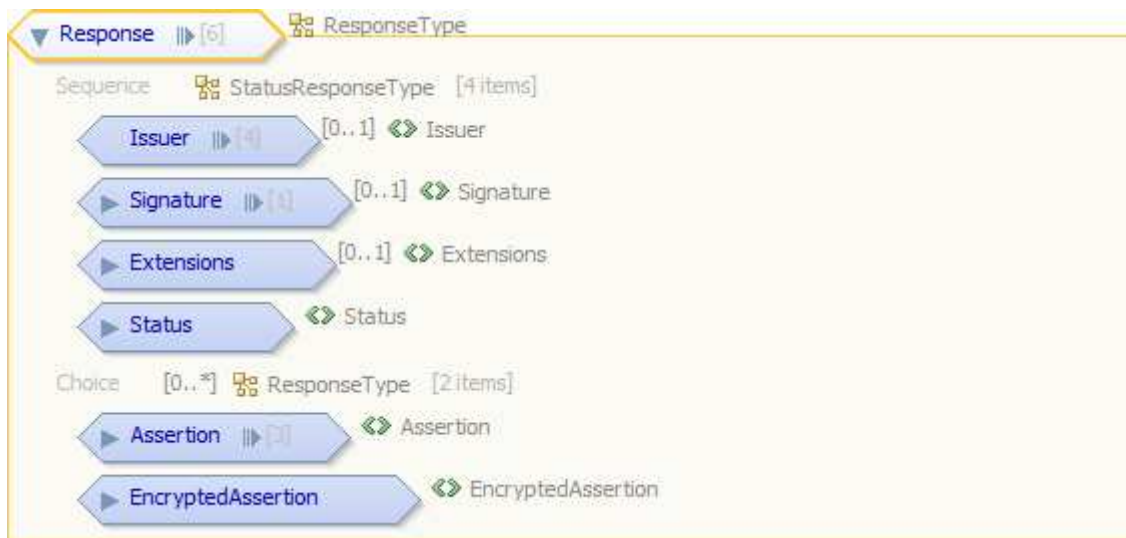


Figure 7: SAML Response

5.2. verifySessionInformation Operation

The mandatory verifySessionInformation operation determines whether the given SAML Assertion (Ticket) is valid or not.

A request to perform the verifySessionInformation operation shall include the parameters listed and defined in Table 3. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	None			
Post conditions	None			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	SAML AssertionDocument	mandatory	Contains the SAML assertion which represents the session information.
Returns	Type		Description	
	verifySessionInformation Response		True if the identity is valid or false otherwise; it contains a status code.	
Throws	Type		Cause	

Table 3: Specification of the verifySessionInformation Operation

The formal platform-specific specification of the verifySessionInformation operation can be found in Appendix A: XML Schema and WSDL Documents.

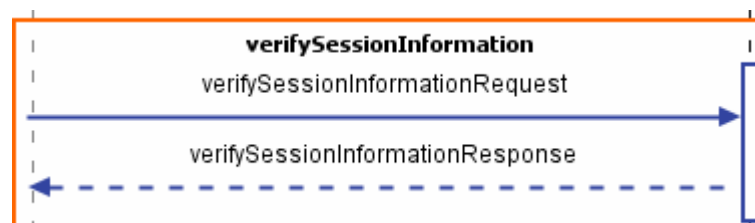


Figure 8: verifySessionInformation operation

5.2.1 verifySessionInformationRequest Type

The verifySessionInformationRequest contains a SAML Assertion (Ticket) which is defined in [OASIS-SAML].

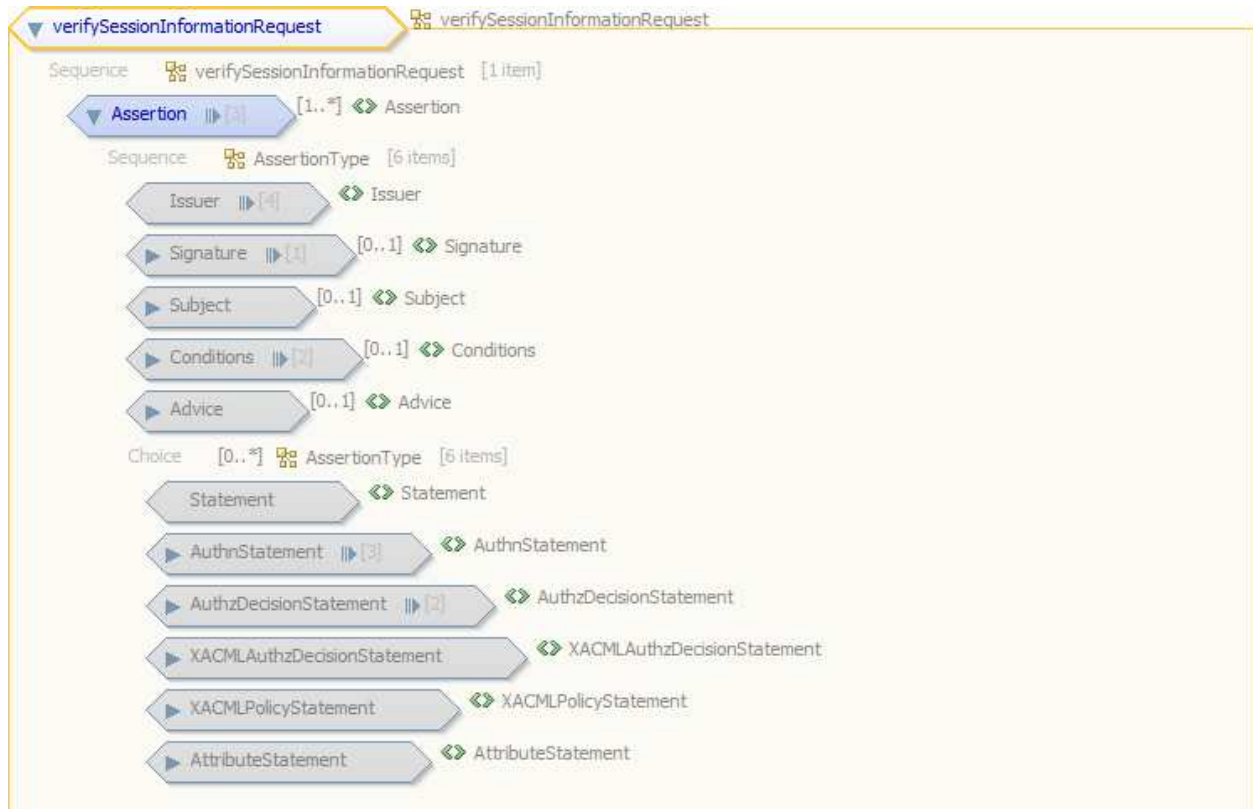


Figure 9: verifySessionInformationRequest

5.3. verifySessionInformationResponse Type

The verifySessionInformationResponse Type consists of the following elements:

- status: Indicates whether verification could successfully be performed or not.
- allValid: Indicates whether the sessionInformation is valid or not.

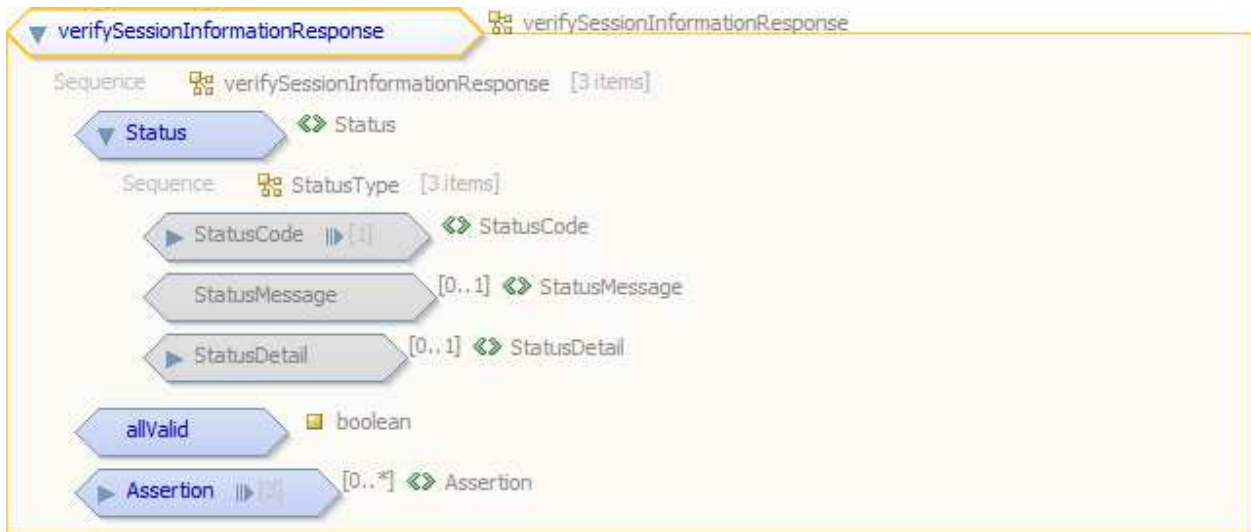


Figure 10: verifySessionInformationResponse

In addition to the status the response of the verifySessionInformation contains all assertions that could be successfully validated.

6. Specification of the Identity Management Interface

The Identity Management Interface is an adaptation of the former ORCHESTRA AuthenticationManagement Interface. It defines operations to create and maintain identities and groups (of identities) as a special kind of an identity.

6.1. activateIdentity Operation

The mandatory activateIdentity operation activates an existing identity. Only active identities can be authenticated.

A request to perform the activateIdentity operation shall include the parameters listed and defined in Table 4. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	Identity exists			
Post conditions	Identity is in state activated (can be authenticated).			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	activateIdentity Request	mandatory	It contains the identity to be activated.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFound Exception		The identity to act on cannot be found.	

	OA_PermissionDenied Exception	The service requestor does not have permissions needed to perform the requested operation.
--	----------------------------------	--

Table 4: Specification of the activateIdentity Operation

The formal platform-specific specification of the activateIdentity operation can be found in Appendix A: XML Schema and WSDL Documents.



Figure 11: activateIdentity operation

6.1.1 activateIdentityRequest Type

The activateIdentityRequest Type consists of the following elements:

- Identity: Represents the identity which shall be activated. As the identities ID is sufficient to uniquely identify it within an Identity Management and Authentication Service instance, it is not required that this object is an instance of a specific identity type (e.g. AttributedIdentity Type).

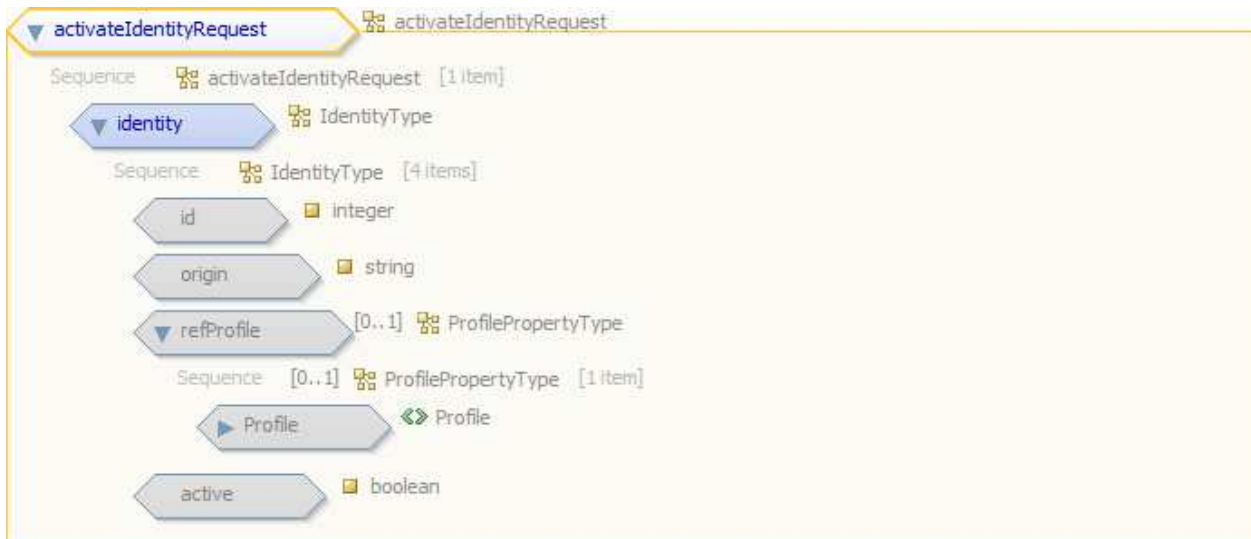


Figure 12: activateIdentityRequest

6.2. deactivateIdentity Operation

The mandatory deactivateIdentity operation deactivates an existing identity. Only active identities can be authenticated.

A request to perform the deactivateIdentity operation *shall* include *the* parameters listed and defined in Table 5. This table also specifies the data type (Type), the obligation [optional|Mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	Identity exists.			
Post conditions	Identity is in deactivated state (it can not be authenticated)			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	deactivateIdentity Request	mandatory	It contains the identity to be deactivated.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFound Exception		The identity to act on cannot be found.	
	OA_PermissionDenied Exception		The service requestor does not have permissions needed to perform the requested operation.	

Table 5: Specification of the deactivateIdentity Operation

The formal platform-specific specification of the deactivateIdentity operation can be found in Appendix A: XML Schema and WSDL Documents.

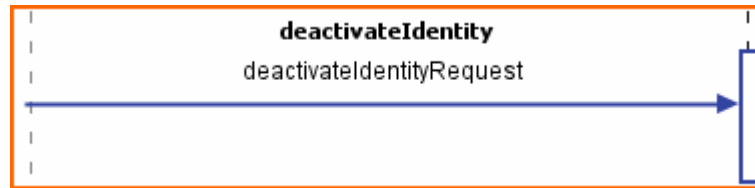


Figure 13: deactivateIdentity operation

6.2.1 deactivateIdentityRequest Type

The deactivateIdentityRequest Type consists of the following elements:

- identity: Represents the identity which shall be deactivated. As the identities ID is sufficient to uniquely identify it within the service instance, it is not required that this object is an instance of a specific identity type (e.g. AttributedIdentity Type).

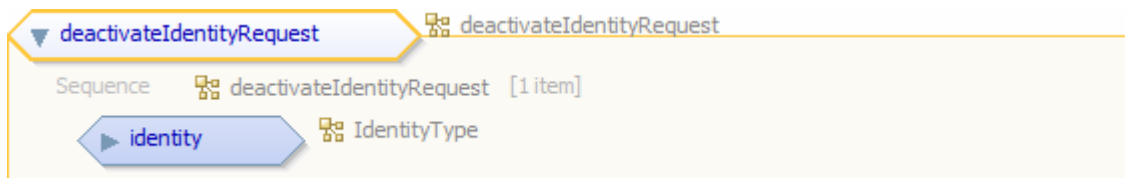


Figure 14: deactivateIdentityRequest

6.3. createIdentity Operation

The mandatory createIdentity operation creates a new identity. The parameter is an identity representation that is specific to the used authentication mechanism. In the present specification UsernameIdentities and GroupIdentities are supported.

A request to perform the createIdentity operation shall include the parameters listed and defined in Table 6. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	None			
Post conditions	The new identity is created.			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	createIdentity Request	mandatory	It contains the identity to be created.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_PermissionDenied Exception		The service requestor does not have permissions needed to perform the requested operation.	

Table 6: Specification of the createIdentity Operation

The formal platform-specific specification of the createIdentity operation can be found in Appendix A: XML Schema and WSDL Documents.



Figure 15: createIdentity operation

6.3.1 createIdentityRequest Type

The createIdentityRequest Type consists of the following elements:

- identity: Represents the identity which shall be created. The attributes inherited from the base type Identity (id and origin) don't have to be set, as they will be overridden by the Identity Management and Authentication Service instance after the identity has been successfully been created. Any other attributes that are set will be stored. The current specification supports the UsernameIdentity Type and requires at least the attribute "username" to be set.

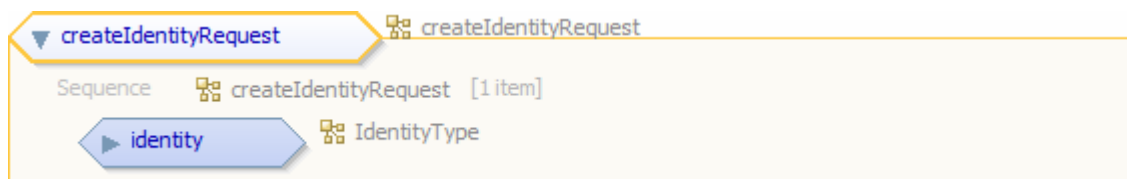


Figure 16: createIdentityRequest

The UsernameIdentity Type extends the AttributedIdentity Type and represents an identity that is identified by a unique username and can be validated by username / password authentication.

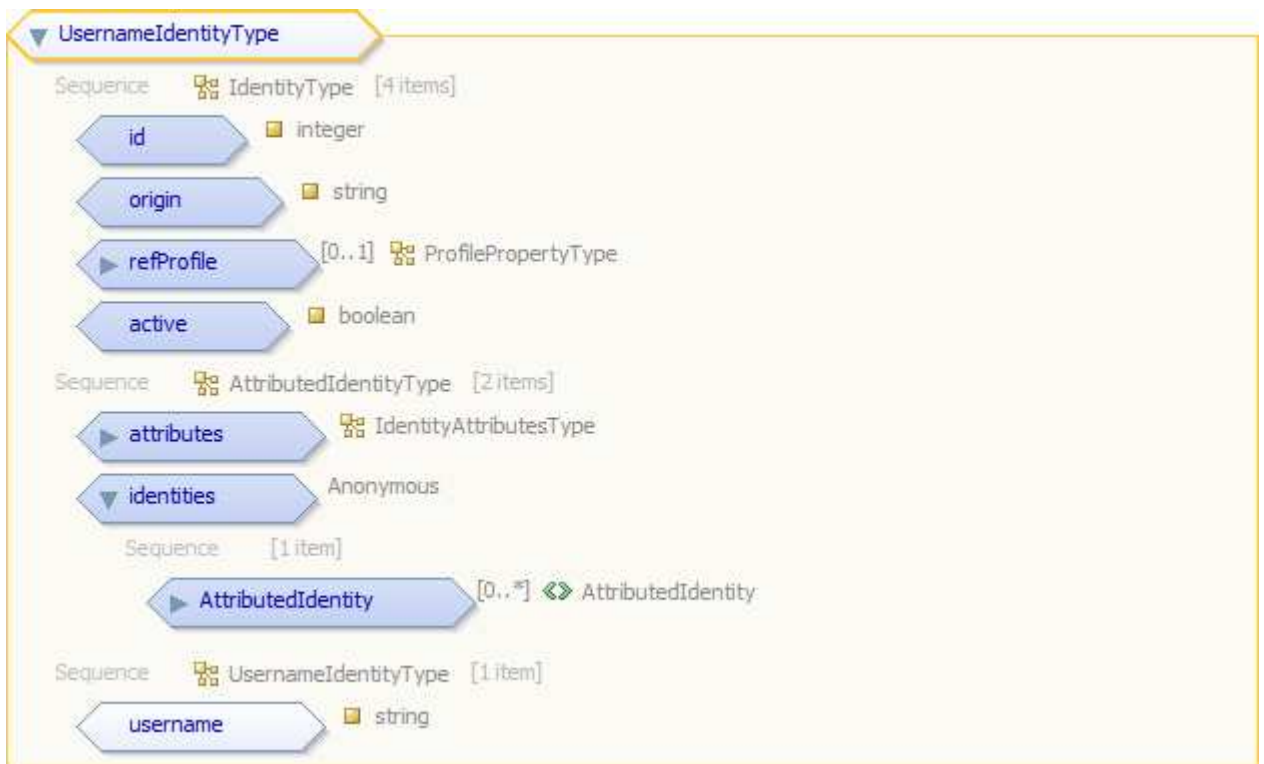


Figure 17: UsernameIdentity Type

The UsernameIdentity Type extends the KeyVectorIdentityAttributes Type consists of the following elements:

- id: unique id, assigned by the service instance
- origin: url of the service instance, also assigned by the service instance
- refProfile list of profiles to which this identity is associated
- active: wheatear the identity may be used for authentication or not
- attributes: key vector pair attributes, e.g. LDAP attributes
- identities: list of group identities to which this identity is associated
- username: unique username

The AttributedIdentity Type allows to attach arbitrary attributes to an identity. Attributes can already be set during creation of a new identity.

Those attributes can be used during an authorisation process to determine whether this identity is allowed to take a certain action or not.



Figure 18: AttributedIdentity Type

The KeyVectorIdentityAttributes Type allows to associate multiple values to an attribute that is identified by a unique key. Based on this an attribute-based role concept can be established where a single identity can act under multiple roles.



Figure 19: KeyVectorIdentityAttributes Type

6.4. deleteIdentity Operation

The mandatory deleteIdentity operation deletes an existing identity. The identity representation is specific to the authentication mechanism used. The current specification of the Identity Management and Authentication Service supports username identities. To ensure consistency the deletion of an identity should be performed within a transaction. In this transaction all references to the identity should also be deleted (e.g. group memberships, ...). A client that invokes the deleteIdentity operation has also to ensure that all profiles referring to the deleted identity are updated accordingly in the appropriate ProfileManagementService instance.

A request to perform the deleteIdentity operation shall include the parameters listed and defined in Table 7. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	Identity exists.			
Post conditions	The identity is deleted and does no longer exist.			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	deleteIdentity Request	mandatory	It contains the identity to be deleted.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFound Exception		The identity to act on cannot be found.	

	OA_PermissionDenied Exception	The service requestor does not have permissions needed to perform the requested operation.
--	----------------------------------	--

Table 7: Specification of the deleteIdentity Operation

The formal platform-specific specification of the deleteIdentity operation can be found in Appendix A: XML Schema and WSDL Documents.



Figure 20: deleteIdentity operation

6.4.1 deleteIdentityRequest Type

The deleteIdentityRequest Type consists of the following elements:

- identity: Represents the identity which shall be deleted. As the identities ID is sufficient to uniquely identify the identity within the service instance, it is not required that this object is an instance of a specific identity type (e.g. AttributedIdentity Type). Instead an object of the base type Identity, having its ID attribute set properly is sufficient.

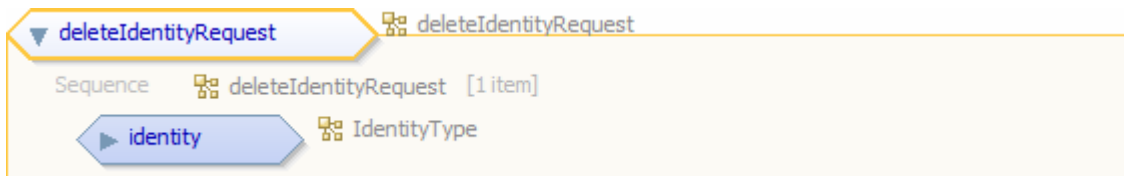


Figure 21: deleteIdentityRequest

6.5. updateIdentity Operation

The mandatory updateIdentity operation updates an existing identity. This operation can also be used to assign identities to a certain group.

A request to perform the updateIdentity operation shall include the parameters listed and defined in Table 8. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	The identity exists.			
Post conditions	The identity is updated.			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	updateIdentity Request	mandatory	It contains the identity to be updated.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFound Exception		The identity to act on cannot be found.	
	OA_PermissionDenied Exception		The service requestor does not have permissions needed to perform the requested operation.	

Table 8: Specification of the updateIdentity Operation

The formal platform-specific specification of the updateIdentity operation can be found in Appendix A: XML Schema and WSDL Documents.

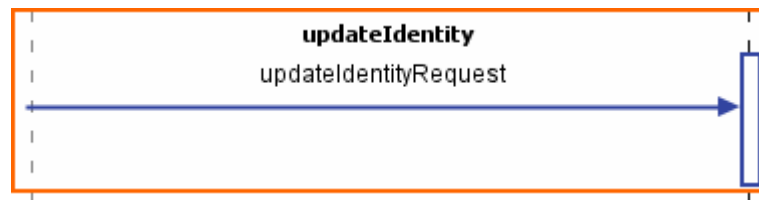


Figure 22: updateIdentity operation

6.5.1 updateIdentityRequest Type

The updateIdentityRequest Type consists of the following elements:

- identity: Represents the identity which shall be updated. The object must be an instance of the type that has been used during creation time. Currently the UsernameIdentity Type is supported. Any attributes being not set are expected to be removed. Any attributes being set, are expected to represent the updated values. To update an identity one may first retrieve an instance of the identity by the “getIdentities” operation, update the correspondent identity object and then use this object in a “updateIdentity” request.

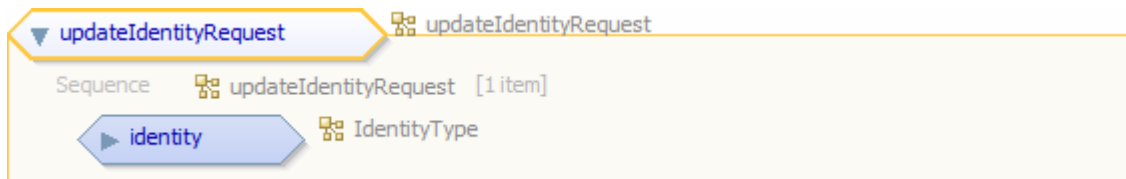


Figure 23: updateIdentityRequest

6.6. addCredentials Operation

The mandatory addCredentials operation adds credentials to a certain identity. Credentials are specific to the authentication mechanism used. The current specifications support username / password credentials that can be added to UsernameIdentities.

A request to perform the addCredentials operation shall include the parameters listed and defined in Table 9. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	The identity exists.			
Post conditions	The identity is associated with the specified credential object.			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	addCredentialsRequest	mandatory	It contains the identity to be updated and the credentials to be assigned with this identity.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFoundException		The identity to act on cannot be found.	
	OA_PermissionDeniedException		The service requestor does not have permissions needed to perform the requested operation.	

Table 9: Specification of the addCredentials Operation

The formal platform-specific specification of the addCredentials operation can be found in Appendix A: XML Schema and WSDL Documents.

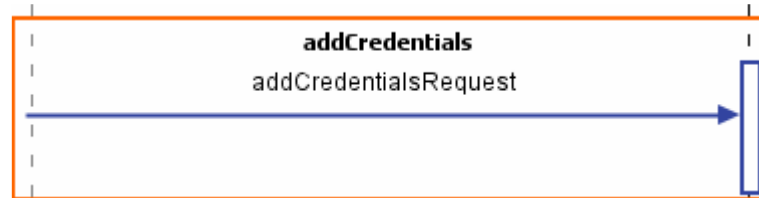


Figure 24: addCredentials operation

6.6.1 addCredentialsRequest Type

The addCredentialsRequest Type consists of the following elements:

- **identity:** Represents the identity to which the credential shall be assigned. As the identities ID is sufficient to uniquely identify the identity within the service instance, it is not required that this object is an instance of a specific type. Instead an object of the base type Identity, having its ID attribute set properly, is sufficient. However, internally this id must refer to a UsernameIdentity object. Trying to add credentials to a GroupIdentity object will result in an exception.
- **credential:** The credentials object which is to be assigned to the identity. The object is expected to be an instance of a concrete type (derived from the base type Credential) having all required attributes set properly. The current specification supports the PasswordCredential type.

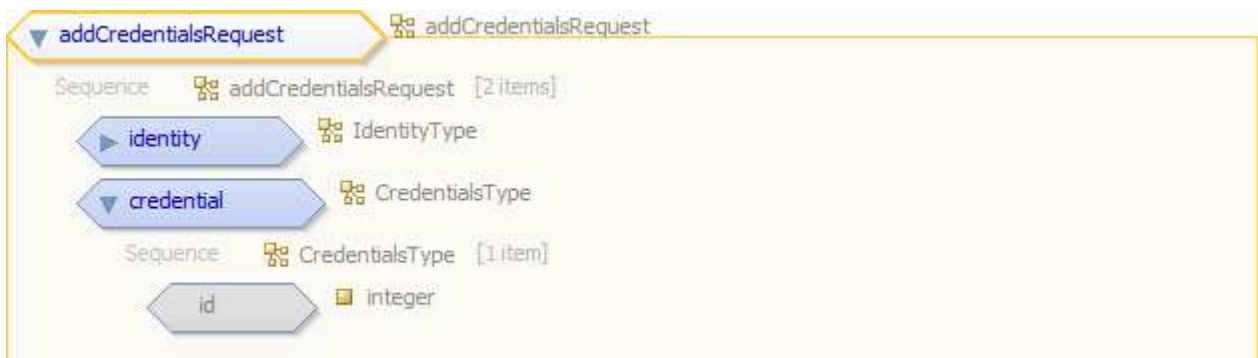


Figure 25: addCredentialsRequest

The PasswordCredentials Type can be used in an username / password authentication scenario.



Figure 26: PasswordCredentials Type

6.7. updateCredentials Operation

The mandatory updateCredentials operation updates credentials for a certain identity.

A request to perform the updateCredentials operation shall include the parameters listed and defined in Table 10. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	None			
Post conditions	None			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	updateCredentialsRequest	mandatory	It contains the identity and the credential to be updated.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_PermissionDeniedException		The service requestor does not have permissions needed to perform the requested operation.	

Table 10: Specification of the updateCredentials Operation

The formal platform-specific specification of the updateCredentials operation can be found in Appendix A: XML Schema and WSDL Documents.

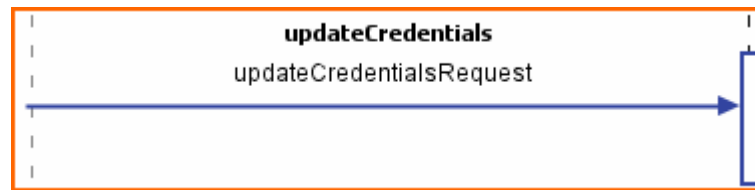


Figure 27: updateCredentials operation

6.7.1 updateCredentialsRequest Type

The updateCredentialsRequest Type consists of the following elements:

- credential: The credential object which replaces the current one. The object is expected to be an instance of a concrete type (a type derived from the base type Credential) having all required attributes set properly. The current specification supports the PasswordCredential Type, which requires the “password” attribute to be set.
- identity: Represents the identity for which the credential to be updated is associated. As the identities ID is sufficient to uniquely identify the identity within the service instance, it is not required that this object is an instance of a specific type. Instead an object of the base type Identity, having its ID attribute set properly is sufficient.

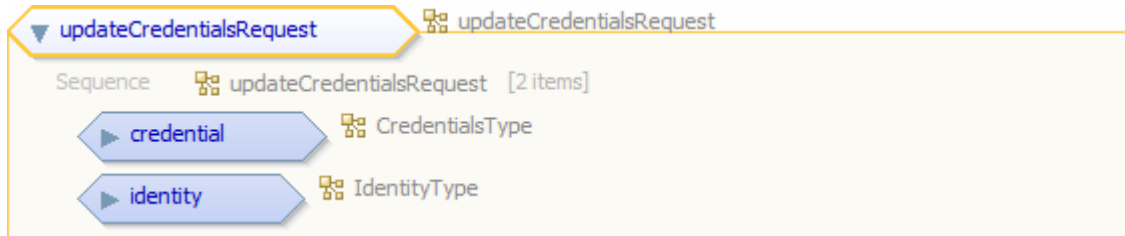


Figure 28: updateCredentialsRequest

6.8. deleteCredentials Operation

The optional deleteCredentials operation removes credentials from a given identity.

A request to perform the deleteCredentials operation shall include the parameters listed and defined in Table 11. This table also specifies the data type (Type), the obligation [optional|mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	The specified identity exists and it is associated with the specified credential object.			
Post conditions	The association is removed. The identity can no longer be authenticated.			
Use	Optional			
Receives	Name	Type	Use	Description
	request	deleteCredentialsRequest	mandatory	It contains the identity from which the credentials are to be deleted.
Returns	Type		Description	
	Not applicable		Not applicable	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_IdentityNotFoundException		The identity to act on cannot be found.	
	OA_PermissionDeniedException		The service requestor does not have permissions needed to perform the requested operation.	

Table 11: Specification of the deleteCredentials Operation

The formal platform-specific specification of the deleteCredentials operation can be found in Appendix A: XML Schema and WSDL Documents.

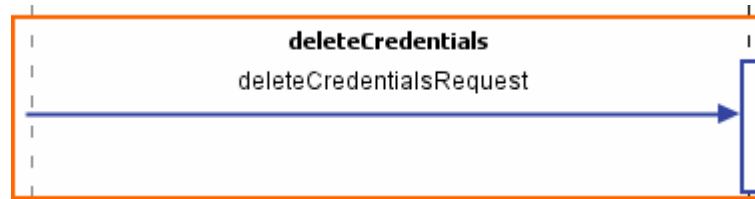


Figure 29: deleteCredentials operation

6.8.1 deleteCredentialsRequest Type

The deleteCredentialsRequest Type consists of the following elements:

- **identity:** Represents the identity from which the credential shall be removed. As the identities ID is sufficient to uniquely identify the identity within the service instance, it is not required that this object is an instance of a specific type. Instead an object of the base type Identity having its ID attribute set properly is sufficient.

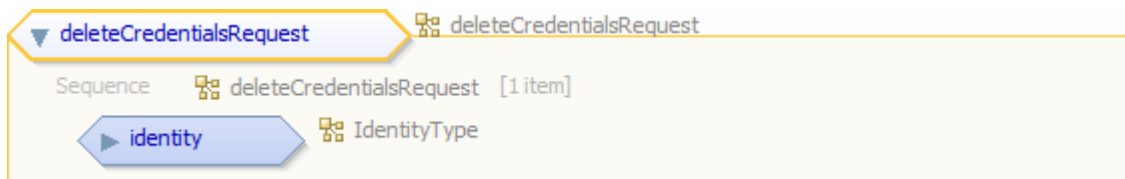


Figure 30: deleteCredentialsRequest

6.9. getIdentitiesOperation

The Mandatory getIdentities operation allows the retrieval of a list of identities specified by a given query. The current specification of the Identity Management Interface does not define a specific query language therefore it is expected that the getIdentities operations always returns all identities maintained by the service instance.

A request to perform the getIdentitiesoperation shall include the parameters listed and defined in Table 8. This table also specifies the data type (Type), the obligation [optional|Mandatory] (Use) and a short description (Description) of each listed parameter. Furthermore the “Description” shall state the consequences for service instances if the correspondent parameter is optional and omitted. Although some values listed in the “Name” column appear to contain spaces, they shall not contain spaces.

Compliance	Not applicable			
Overrides	Not applicable			
Preconditions	None			
Post conditions	None			
Use	Mandatory			
Receives	Name	Type	Use	Description
	request	getIdentities Request	Mandatory	It contains an optional query.
Returns	Type		Description	
	SequenceOfIdentityType		Set of identities matching the specified query.	
Throws	Type		Cause	
	OA_InvalidParameterValue		Operation request contains an invalid parameter value. Returns the name of the parameter with invalid value.	
	OA_MissingParameterValue		Operation request does not include a parameter value. Returns the name of the missing parameter.	
	OA_NoApplicableCode		No other basic or service-specific exception type applies.	
	OA_InternalError		A problem occurred in the runtime environment (e.g. out of memory).	
	OA_PermissionDenied Exception		The service requestor does not have permissions needed to perform the requested operation.	

Table 12: Specification of the getIdentitiesOperation

The formal platform-specific specification of the `getIdentities` operation can be found in Appendix A: XML Schema and WSDL Documents.

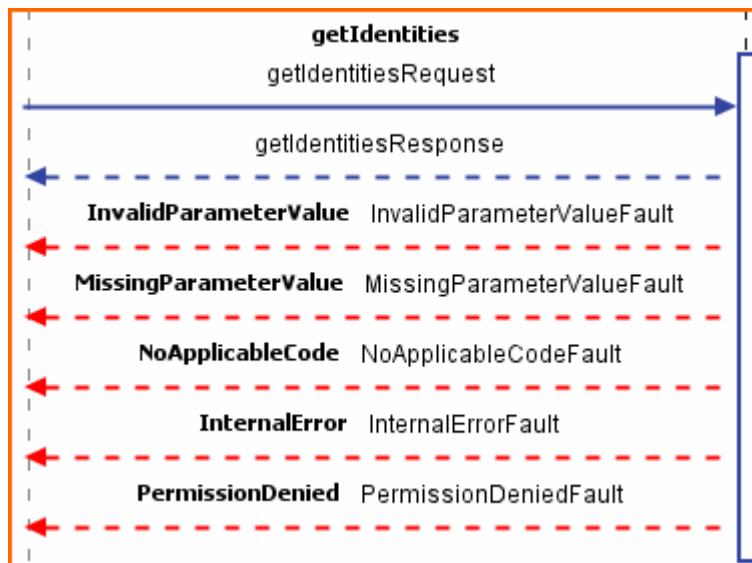


Figure 31: `getIdentities` operation

6.9.1 `getIdentitiesRequest` Type

The `getIdentitiesRequest` Type consists of the following elements:

- `query`: A query used to specify a subset of identities to return. The current specification does not define a query language and thus expects the `getIdentities` operation to always return all identities maintained by the service instance.

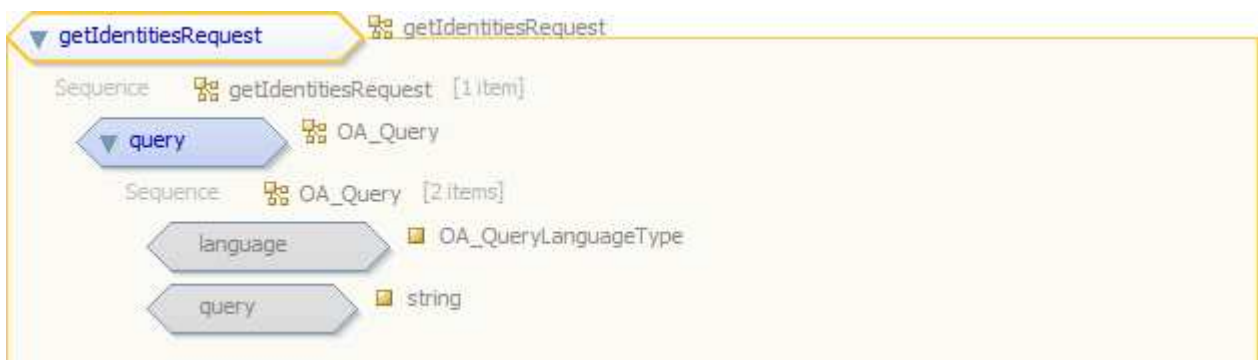


Figure 32: `getIdentitiesRequest`

6.9.2 `getIdentitiesResponse` Type

The `getIdentitiesResponse` Type consists of the following elements:

- `identities`: This object consists of a sub-element “Sequence” containing all the identities currently stored on the service instance. It may contain identities of different types (e.g. `GroupIdentities` and `UsernameIdentities`) that are derived from the generic `Identity` Type.

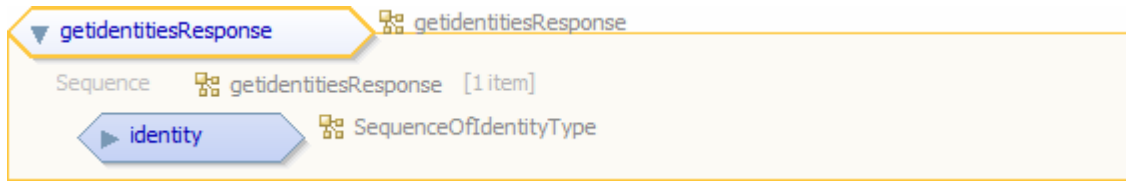


Figure 33: `getIdentitiesResponse`

7. References

- OASIS-SAML** OASIS Standard: Security Assertion Markup Language (SAML) v2.0, published 03/2005, available from:
<<http://www.oasis-open.org/specs/index.php#samlv2.0>>
- ORCH-Authentication-ImplSpec** Implementation Specification of the Authentication Service (Version 1.1), ORCHESTRA Consortium, Editor: Environmental Informatics Group (EIG), available from:
<<http://www.eu-orchestra.org/publications.shtml#OAImplspecs>>
- ORCH-Authentication-AbstractSpec** Service Specification of the Authentication Service (Version 1.3), ORCHESTRA Consortium, Editor: Environmental Informatics Group (EIG), available from:
<<http://www.eu-orchestra.org/publications.shtml#OASpecs>>
- ORCH-User-Mgmt-AbstractSpec** Service Specification of the User Management Service (Version 1.8), ORCHESTRA Consortium, Editor Environmental Informatics Group (EIG), available from:
<<http://www.eu-orchestra.org/publications.shtml#OASpecs>>
- SANY D2.3.3, 2009** SANY D2.3.3 “Specification of the Sensor Service Architecture V2”, SANY IP document; available from sany website:
<<http://sany-ip.eu/biblio>>
- SANY-D2.4.3, 2009** SANY D2.4.3 “Sensor Service Specification V2”, SANY IP document; available from SANY website:
<<http://sany-ip.eu/biblio>>
- SANY-Authorisation** Specification of the Policy Management and Authorisation Service (Version 1.1), SANY Consortium, Editor: Environmental Informatics Group (EIG), available from SANY website:
<<http://www.sany-ip.eu/biblio>>
- SANY-PEP** Specification of the Policy Enforcement Service (Version 1.1), SANY Consortium, Editor: Environmental Informatics Group (EIG), available from SANY website:
<<http://www.sany-ip.eu/biblio>>

SANY-ProfileManagement

Specification of the Profile Management Service (Version 1.1), SANY Consortium, Editor: Environmental Informatics Group (EIG), available from SANY website:

<<http://www.sany-ip.eu/biblio>>

8. Appendix A: XML Schema and WSDL Documents

8.1. XML Schema Documents

The following XML Schema Documents define the data types of this service.

The XML schema documents of the used data types are bundled in a zip file with the present document and can be downloaded at

<http://repository.sany-ip.eu/svn/schemas/v2/schema/security/> and from the SANY website

<http://www.sany-ip.eu/>.

In addition to XML Schema Documents specified in this appendix, this specification requires several normative XML Schema Documents. These XML Schema Documents define the common data types, e.g. common Exception Types, Basic Data Types and the GML Profile.

This file contains the XML Schema documents of the OA Basic Service and Identity Management and Authentication Service.

The namespaces

<http://eu-orchestra.org/OA/OABasicService/types/1.0>,

<http://eu-orchestra.org/OA/OABasicService/exceptions/1.0>, are used.

8.1.1 identity_authen_exceptions.xsd

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"

xmlns:ia_exc="http://www.enviromatics.net/WS/IdentityManagementAndAuthenticat
ionService/exceptions/2.0"
xmlns:oab_exc="http://eu-
orchestra.org/OA/OABasicService/exceptions/1.0"
elementFormDefault="qualified"

targetNamespace="http://www.enviromatics.net/WS/IdentityManagementAndAuthenti
cationService/exceptions/2.0"
version="1.0">
<import namespace="http://eu-
orchestra.org/OA/OABasicService/exceptions/1.0"

schemaLocation="http://www.enviromatics.net/WS/OrchestraArchive/oa_basic_ex.x
sd"/>

<element name="IdentityNotFoundException"
substitutionGroup="oab_exc:OA_AbstractException"
type="ia_exc:IdentityNotFoundExceptionType"/>
<complexType name="IdentityNotFoundExceptionType">
<complexContent>
<extension base="oab_exc:OA_AbstractException">
<sequence/>
```

```

        </extension>
      </complexContent>
    </complexType>
    <complexType name="IdentityNotFoundExceptionPropertyType">
      <sequence minOccurs="0">
        <element ref="ia_exc:IdentityNotFoundException"/>
      </sequence>
    </complexType>

    <element name="AuthenticationFailedException"
      substitutionGroup="oab_exc:OA_AbstractException"
      type="ia_exc:AuthenticationFailedExceptionType"/>
    <complexType name="AuthenticationFailedExceptionType">
      <complexContent>
        <extension base="oab_exc:OA_AbstractException">
          <sequence/>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="AuthenticationFailedExceptionPropertyType">
      <sequence minOccurs="0">
        <element ref="ia_exc:AuthenticationFailedException"/>
      </sequence>
    </complexType>
  </schema>

```

8.1.2 identity_authen__types.xsd

```

<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"

xmlns:ia_types="http://www.enviomatics.net/WS/IdentityManagementAndAuthentic
ationService/types/2.0"

xmlns:pms_types="http://www.enviomatics.net/WS/ProfileManagementService/type
s/2.0"

targetNamespace="http://www.enviomatics.net/WS/IdentityManagementAndAuthenti
cationService/types/2.0"
    elementFormDefault="qualified" version="1.0">
    <import
namespace="http://www.enviomatics.net/WS/ProfileManagementService/types/2.0"

schemaLocation="http://www.enviomatics.net/WS/ProfileManagementService/types
/2.0/profilemanagement_types.xsd"/>

    <element name="AuthenticatedIdentity"
type="ia_types:AuthenticatedIdentityType"
substitutionGroup="ia_types:AttributedIdentity"/>
    <complexType name="AuthenticatedIdentityType">
    <annotation>
    <documentation>Represents a identity after it is authenticated.
An authenticated identity consists of the authenticated identity as well as a
time coverage the authentication is valid.</documentation>
    </annotation>
    <complexContent>
    <extension base="ia_types:AttributedIdentityType">
    <sequence>
    <element name="sessionId" type="string"/>
    <element name="validityEnd" type="dateTime"/>
    </sequence>
    </extension>
    </complexContent>
    </complexType>
    <complexType name="AuthenticatedIdentityPropertyType">
    <sequence minOccurs="0">
    <element ref="ia_types:AuthenticatedIdentity"/>
    </sequence>
    </complexType>

    <element name="AttributedIdentity" type="ia_types:AttributedIdentityType"
substitutionGroup="ia_types:Identity"/>
    <complexType name="AttributedIdentityType">
    <complexContent>
    <extension base="ia_types:IdentityType">
    <sequence>
    <element name="attributes"
type="ia_types:IdentityAttributesType"/>
    <element name="identities">
    <complexType>
    <sequence>
    <element ref="ia_types:AuthenticatedIdentity"
minOccurs="0" maxOccurs="unbounded"/>

```

```

        </sequence>
      </complexType>
    </element>
  </sequence>
</extension>
</complexContent>
</complexType>

<element name="IdentityAttributes"
type="ia_types:IdentityAttributesType"/>
<complexType name="IdentityAttributesType">
  <sequence/>
</complexType>
<complexType name="IdentityAttributesPropertyType">
  <sequence minOccurs="0">
    <element ref="ia_types:IdentityAttributes"/>
  </sequence>
</complexType>

<element name="KeyVectorIdentityAttributes"
type="ia_types:KeyVectorIdentityAttributesType"
substitutionGroup="ia_types:IdentityAttributes"/>
<complexType name="KeyVectorIdentityAttributesType">
  <complexContent>
    <extension base="ia_types:IdentityAttributesType">
      <sequence>
        <element ref="ia_types:KeyVectorPair" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="KeyVectorPair" type="ia_types:KeyVectorPairType"/>
<complexType name="KeyVectorPairType">
  <sequence>
    <element name="key" type="string"/>
    <element name="vector">
      <complexType>
        <sequence>
          <element name="element" type="string" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
      </complexType>
    </element>
  </sequence>
</complexType>

<element name="UsernameIdentity" type="ia_types:UsernameIdentityType"
substitutionGroup="ia_types:AttributedIdentity"/>
<complexType name="UsernameIdentityType">
  <complexContent>
    <extension base="ia_types:AttributedIdentityType">
      <sequence>
        <element name="username" type="string"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name="UsernameIdentityPropertyType">

```

```

    <sequence minOccurs="0">
      <element ref="ia_types:UsernameIdentity"/>
    </sequence>
  </complexType>

  <element name="GroupIdentity" type="ia_types:GroupIdentityType"
substitutionGroup="ia_types:AttributedIdentity"/>
  <complexType name="GroupIdentityType">
    <complexContent>
      <extension base="ia_types:AttributedIdentityType">
        <sequence>
          <element name="groupname" type="string"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="GroupIdentityPropertyType">
    <sequence minOccurs="0">
      <element ref="ia_types:GroupIdentity"/>
    </sequence>
  </complexType>

  <element name="PasswordCredentials"
type="ia_types:PasswordCredentialsType"
substitutionGroup="ia_types:Credentials"/>
  <complexType name="PasswordCredentialsType">
    <annotation>
      <documentation>Represents the identities password. The password
gets stored encoded only and is transfered using Base64-encoding. Using
Base64-Encoding prevents the occurance of not valid characters within the XML
documents.</documentation>
    </annotation>
    <complexContent>
      <extension base="ia_types:CredentialsType">
        <sequence>
          <element name="password" type="string"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="PasswordCredentialsPropertyType">
    <sequence minOccurs="0">
      <element ref="ia_types:PasswordCredentials"/>
    </sequence>
  </complexType>

  <element name="Identity" type="ia_types:IdentityType"/>
  <complexType name="IdentityType">
    <sequence>
      <element name="id" type="integer"/>
      <element name="origin" type="string"/>
      <!--element name="refGroups">
        <complexType>
          <sequence>
            <element name="Sequence">
              <complexType>
                <sequence>
                  <element name="element"
type="pms_types:GroupPropertyType" minOccurs="0" maxOccurs="unbounded"/>
                </sequence>
              </complexType>
            </element>
          </sequence>
        </complexType>
      </-->
    </sequence>
  </complexType>

```

```

        </complexType>
      </element>
    </sequence>
  </complexType>
</element-->
  <element name="refProfile" type="pms_types:ProfilePropertyType"
minOccurs="0"/>
  <element name="active" type="boolean"/>
</sequence>
</complexType>
<complexType name="IdentityPropertyType">
  <sequence minOccurs="0">
    <element ref="ia_types:Identity"/>
  </sequence>
</complexType>

  <element name="Credentials" type="ia_types:CredentialsType"/>
<complexType name="CredentialsType">
  <annotation>
    <documentation>Credentials e.g. used by identities in the context
of Authentication Services.</documentation>
  </annotation>
  <sequence>
    <element name="id" type="integer"/>
  </sequence>
</complexType>
<complexType name="CredentialsPropertyType">
  <sequence minOccurs="0">
    <element ref="ia_types:Credentials"/>
  </sequence>
</complexType>

  <element name="AuthenticationSessionInformation"
type="ia_types:AuthenticationSessionInformationType"/>
<complexType name="AuthenticationSessionInformationType">
  <annotation>
    <documentation>Identifies a session and is used to determine
whether the by it specified session is still valid. What shape does a
SessionKey have?</documentation>
  </annotation>
  <sequence>
    <element name="authenticatedIdentities">
      <complexType>
        <sequence>
          <element name="Sequence">
            <complexType>
              <sequence>
                <element name="element"
type="ia_types:AuthenticatedIdentityPropertyType" minOccurs="0"
maxOccurs="unbounded"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
      </complexType>
    </element>
  </sequence>
</complexType>
<complexType name="AuthenticationSessionInformationPropertyType">
  <sequence minOccurs="0">

```

```

        <element ref="ia_types:AuthenticationSessionInformation"/>
    </sequence>
</complexType>

<element name="SequenceOfIdentity"
type="ia_types:SequenceOfIdentityType"/>
<complexType name="SequenceOfIdentityType">
    <sequence>
        <element name="identities">
            <complexType>
                <sequence>
                    <element name="Sequence">
                        <complexType>
                            <sequence>
                                <element name="Element"
type="ia_types:IdentityPropertyType" minOccurs="0" maxOccurs="unbounded"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
            </complexType>
        </element>
    </sequence>
</complexType>
<complexType name="SequenceOfIdentityPropertyType">
    <sequence minOccurs="0">
        <element ref="ia_types:SequenceOfIdentity"/>
    </sequence>
</complexType>
</schema>

```

8.1.3 identity_authen_requests.xsd

```

<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:gml="http://www.opengis.net/gml"

  xmlns:ia_types="http://www.enviromatics.net/WS/IdentityManagementAndAuthentic
ationService/types/2.0"

  xmlns:ia_requests="http://www.enviromatics.net/WS/IdentityManagementAndAuthen
ticationService/requests/2.0"
    xmlns:oas="http://eu-orchestra.org/OA/OABasicService/types/1.0"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:samla="urn:oasis:names:tc:SAML:2.0:assertion"

  targetNamespace="http://www.enviromatics.net/WS/IdentityManagementAndAuthenti
cationService/requests/2.0"
    elementFormDefault="qualified"
    version="1.0">
  <import
namespace="http://www.enviromatics.net/WS/IdentityManagementAndAuthentication
Service/types/2.0"

schemaLocation="http://www.enviromatics.net/WS/IdentityManagementAndAuthentic
ationService/types/2.0/identity_authen_types.xsd" />
  <!-- TODO: samp protocol cannot be verified because of xmlenc and xmldsig
-->
  <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
    schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-assertion-2.0.xsd" />
  <import namespace="urn:oasis:names:tc:SAML:2.0:protocol"
    schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-protocol-2.0.xsd" />
  <import namespace="http://eu-orchestra.org/OA/OABasicService/types/1.0"

schemaLocation="http://www.enviromatics.net/WS/OrchestraArchive/oa_basic.xsd"
/>

  <element name="activateIdentityRequest"
type="ia_requests:activateIdentityRequest" />
  <complexType name="activateIdentityRequest">
    <sequence>
      <element name="identity" type="ia_types:IdentityType" />
    </sequence>
  </complexType>

  <element name="addCredentialsRequest"
type="ia_requests:addCredentialsRequest" />
  <complexType name="addCredentialsRequest">
    <sequence>
      <element name="identity" type="ia_types:IdentityType" />
      <element name="credential" type="ia_types:CredentialsType" />
    </sequence>
  </complexType>
  <complexType name="addIdentityToGroupRequest">
    <sequence>
      <element name="identity" type="ia_types:IdentityType" />

```

```

        <element name="group" type="ia_types:GroupIdentityType"/>
    </sequence>
</complexType>
<complexType name="deleteIdentityFromGroupRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
        <element name="group" type="ia_types:GroupIdentityType"/>
    </sequence>
</complexType>

<element name="createIdentityRequest"
type="ia_requests:createIdentityRequest"/>
<complexType name="createIdentityRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

<element name="deactivateIdentityRequest"
type="ia_requests:deactivateIdentityRequest"/>
<complexType name="deactivateIdentityRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

<element name="deleteCredentialsRequest"
type="ia_requests:deleteCredentialsRequest"/>
<complexType name="deleteCredentialsRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

<element name="deleteIdentityRequest"
type="ia_requests:deleteIdentityRequest"/>
<complexType name="deleteIdentityRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

<element name="updateCredentialsRequest"
type="ia_requests:updateCredentialsRequest"/>
<complexType name="updateCredentialsRequest">
    <sequence>
        <element name="credential" type="ia_types:CredentialsType"/>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

<element name="updateIdentityRequest"
type="ia_requests:updateIdentityRequest"/>
<complexType name="updateIdentityRequest">
    <sequence>
        <element name="identity" type="ia_types:IdentityType"/>
    </sequence>
</complexType>

```

```

    <element name="verifySessionInformationRequest"
type="ia_requests:verifySessionInformationRequest"/>
    <complexType name="verifySessionInformationRequest">
        <sequence>
            <element ref="samla:Assertion" maxOccurs="unbounded"/>
        </sequence>
    </complexType>

    <element name="verifySessionInformationResponse"
type="ia_requests:verifySessionInformationResponse"/>
    <complexType name="verifySessionInformationResponse">
        <sequence>
            <element ref="sampl:Status"/>
            <element name="allValid" type="boolean"/>
            <element ref="samla:Assertion" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>

    <element name="getIdentitiesRequest"
type="ia_requests:getIdentitiesRequest"/>
    <complexType name="getIdentitiesRequest">
        <sequence>
            <element name="query" type="oas:OA_Query"/>
        </sequence>
    </complexType>

    <element name="getIdentitiesResponse"
type="ia_requests:getIdentitiesResponse"/>
    <complexType name="getIdentitiesResponse">
        <sequence>
            <element name="identity" type="ia_types:SequenceOfIdentityType"/>
        </sequence>
    </complexType>
</schema>

```

8.2. WSDL Document

The following WSDL Version 1.1 document is the formal specification of the Identity Management and Authentication Service according to rules of the “SANY Web Services Platform”. It defines the mandatory SOAP binding.

The WSDL document is bundled in a zip file with the present document and can be downloaded at

<http://repository.sany-ip.eu/svn/schemas/v2/wSDL/security/> and from the SANY website

<http://www.sany-ip.eu/>.

```
<?xml version="1.0"?>
<wSDL:definitions xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"

  xmlns:ia="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService"

  xmlns:ia_requests="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService/requests/2.0"

  xmlns:ia_exc="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService/exceptions/2.0"

  xmlns:pa_exc="http://www.enviroomatics.net/WS/PolicyManagementAndAuthorisationService/exceptions/2.0"

  xmlns:ia_types="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService/types/2.0"
    xmlns:oab_exc="http://eu-orchestra.org/OA/OABasicService/exceptions/1.0"
    xmlns:oab_types="http://eu-orchestra.org/OA/OABasicService/types/1.0"
    xmlns:bdT="http://eu-orchestra.org/basicTypes/1.0"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:samla="urn:oasis:names:tc:SAML:2.0:assertion"
    name="IdentityManagementAndAuthenticationServiceInstance"

  targetNamespace="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService" xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype">
  <wSDL:types>
    <xs:schema
      targetNamespace="http://www.enviroomatics.net/WS/IdentityManagementAndAuthenticationService"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified"
        attributeFormDefault="qualified">
      <xs:import namespace="http://eu-orchestra.org/basicTypes/1.0"

      schemaLocation="http://www.enviroomatics.net/WS/OrchestraArchive/basic_types.xsd"/>
```

```

        <xs:import namespace="http://eu-
orchestra.org/OA/OABasicService/exceptions/1.0"

schemaLocation="http://www.enviomatics.net/WS/OrchestraArchive/oa_basic_ex.x
sd"/>
        <xs:import namespace="http://eu-
orchestra.org/OA/OABasicService/types/1.0"

schemaLocation="http://www.enviomatics.net/WS/OrchestraArchive/oa_basic.xsd"
/>
        <xs:import
namespace="http://www.enviomatics.net/WS/IdentityManagementAndAuthentication
Service/requests/2.0"

schemaLocation="http://www.enviomatics.net/WS/IdentityManagementAndAuthentic
ationService/requests/2.0/identity_authen_requests.xsd"/>
        <xs:import
namespace="http://www.enviomatics.net/WS/PolicyManagementAndAuthorisationSer
vice/exceptions/2.0"

schemaLocation="http://www.enviomatics.net/WS/PolicyManagementAndAuthorisati
onService/exceptions/2.0/policy_author_exceptions.xsd"/>
        <xs:import
namespace="http://www.enviomatics.net/WS/IdentityManagementAndAuthentication
Service/exceptions/2.0"

schemaLocation="http://www.enviomatics.net/WS/IdentityManagementAndAuthentic
ationService/exceptions/2.0/identity_authen_exceptions.xsd"/>
        <xs:import
namespace="http://www.enviomatics.net/WS/IdentityManagementAndAuthentication
Service/types/2.0"

schemaLocation="http://www.enviomatics.net/WS/IdentityManagementAndAuthentic
ationService/types/2.0/identity_authen_types.xsd"/>
        <xs:import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-assertion-2.0.xsd"/>
        <xs:import namespace="urn:oasis:names:tc:SAML:2.0:protocol"
schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-protocol-2.0.xsd"/>
    </xs:schema>
</wdd:types>
<wdd:message name="getCapabilitiesRequest">
    <wdd:part name="request"
element="oab_types:OA_GetCapabilitiesRequest"/>
</wdd:message>
<wdd:message name="getCapabilitiesResponse">
    <wdd:part name="response"
element="oab_types:OA_GetCapabilitiesResponse"/>
</wdd:message>
<wdd:message name="activateIdentityRequest">
    <wdd:part name="request"
element="ia_requests:activateIdentityRequest"/>
</wdd:message>
<wdd:message name="addCredentialsRequest">
    <wdd:part name="request"
element="ia_requests:addCredentialsRequest"/>
</wdd:message>
<wdd:message name="createIdentityRequest">

```

```

        <wsdl:part name="request"
element="ia_requests:createIdentityRequest"/>
    </wsdl:message>
    <wsdl:message name="deactivateIdentityRequest">
        <wsdl:part name="request"
element="ia_requests:deactivateIdentityRequest"/>
    </wsdl:message>
    <wsdl:message name="loginRequest">
        <wsdl:part name="request" element="samlp:AuthnRequest"/>
    </wsdl:message>
    <wsdl:message name="loginResponse">
        <wsdl:part name="response" element="samlp:Response"/>
    </wsdl:message>
    <wsdl:message name="deleteCredentialsRequest">
        <wsdl:part name="request"
element="ia_requests:deleteCredentialsRequest"/>
    </wsdl:message>
    <wsdl:message name="deleteIdentityRequest">
        <wsdl:part name="request"
element="ia_requests:deleteIdentityRequest"/>
    </wsdl:message>
    <wsdl:message name="updateCredentialsRequest">
        <wsdl:part name="request"
element="ia_requests:updateCredentialsRequest"/>
    </wsdl:message>
    <wsdl:message name="updateIdentityRequest">
        <wsdl:part name="request"
element="ia_requests:updateIdentityRequest"/>
    </wsdl:message>
    <wsdl:message name="verifySessionInformationRequest">
        <wsdl:part name="request"
element="ia_requests:verifySessionInformationRequest"/>
    </wsdl:message>
    <wsdl:message name="verifySessionInformationResponse">
        <wsdl:part name="response"
element="ia_requests:verifySessionInformationResponse"/>
    </wsdl:message>
    <wsdl:message name="getIdentitiesRequest">
        <wsdl:part name="request"
element="ia_requests:getIdentitiesRequest"/>
    </wsdl:message>
    <wsdl:message name="getIdentitiesResponse">
        <wsdl:part name="response" element="ia_types:SequenceOfIdentity"/>
    </wsdl:message>
    <wsdl:message name="InvalidParameterValueFault">
        <wsdl:part name="OA_InvalidParameterValue"
element="oab_exc:OA_InvalidParameterValue"/>
    </wsdl:message>
    <wsdl:message name="MissingParameterValueFault">
        <wsdl:part name="OA_MissingParameterValue"
element="oab_exc:OA_MissingParameterValue"/>
    </wsdl:message>
    <wsdl:message name="NoApplicableCodeFault">
        <wsdl:part name="OA_NoApplicableCode"
element="oab_exc:OA_NoApplicableCode"/>
    </wsdl:message>
    <wsdl:message name="InternalErrorFault">
        <wsdl:part name="OA_InternalError"
element="oab_exc:OA_InternalError"/>
    </wsdl:message>

```

```

    <wsdl:message name="IdentityNotFoundFault">
      <wsdl:part name="IdentityNotFoundException"
element="ia_exc:IdentityNotFoundException"/>
    </wsdl:message>
    <wsdl:message name="PermissionDeniedFault">
      <wsdl:part name="PermissionDeniedException"
element="pa_exc:PermissionDeniedException"/>
    </wsdl:message>
    <wsdl:message name="AuthenticationFailedFault">
      <wsdl:part name="AuthenticationFailedException"
element="ia_exc:AuthenticationFailedException"/>
    </wsdl:message>
    <wsdl:message name="VersionNegotiationFailed">
      <wsdl:part name="OA_VersionNegotiationFailed"
element="oab_exc:OA_VersionNegotiationFailed"/>
    </wsdl:message>
    <wsdl:message name="UnsupportedCapSchema">
      <wsdl:part name="OA_UnsupportedCapSchema"
element="oab_exc:OA_UnsupportedCapSchema"/>
    </wsdl:message>
    <wsdl:portType name="IdentityManagementAndAuthenticationService">
      <wsdl:operation name="getCapabilities">
        <wsdl:input name="capabilitiesRequest"
message="ia:getCapabilitiesRequest"/>
        <wsdl:output name="capabilitiesResponse"
message="ia:getCapabilitiesResponse"/>
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault"/>
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault"/>
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault"/>
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault"/>
        <wsdl:fault name="VersionNegotiationFailed"
message="ia:VersionNegotiationFailed"/>
        <wsdl:fault name="UnsupportedCapSchema"
message="ia:UnsupportedCapSchema"/>
      </wsdl:operation>
      <wsdl:operation name="verifySessionInformation">
        <wsdl:documentation>The verifySessionInformation operation
determines whether the given session-information is
valid.</wsdl:documentation>
        <wsdl:input name="verifySessionInformationRequestRequest"
message="ia:verifySessionInformationRequest"/>
        <wsdl:output name="verifySessionInformationRequestResponse"
message="ia:verifySessionInformationResponse"/>
      </wsdl:operation>
      <wsdl:operation name="updateIdentity">
        <wsdl:input name="updateIdentityRequestRequest"
message="ia:updateIdentityRequest"/>
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault"/>
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault"/>
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault"/>
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault"/>

```

```

        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="updateCredentials">
        <wsdl:input name="updateCredentialsRequestRequest"
message="ia:updateCredentialsRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="deleteIdentity">
        <wsdl:input name="deleteIdentityRequestRequest"
message="ia:deleteIdentityRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="deleteCredentials">
        <wsdl:input name="deleteCredentialsRequestRequest"
message="ia:deleteCredentialsRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="login">
        <wsdl:input name="loginRequestRequest"
message="ia:loginRequest" />
        <wsdl:output name="loginRequestResponse"
message="ia:loginResponse" />
    </wsdl:operation>
    <wsdl:operation name="deactivateIdentity">
        <wsdl:input name="deactivateIdentityRequestRequest"
message="ia:deactivateIdentityRequest" />

```

```

        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="createIdentity">
        <wsdl:input name="createIdentityRequestRequest"
message="ia:createIdentityRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="addCredentials">
        <wsdl:input name="addCredentialsRequestRequest"
message="ia:addCredentialsRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="activateIdentity">
        <wsdl:input name="activateIdentityRequestRequest"
message="ia:activateIdentityRequest" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="IdentityNotFound"
message="ia:IdentityNotFoundFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
    <wsdl:operation name="getIdentities">

```

```

        <wsdl:input name="getIdentitiesRequestRequest"
message="ia:getIdentitiesRequest" />
        <wsdl:output name="getIdentitiesRequestResponse"
message="ia:getIdentitiesResponse" />
        <wsdl:fault name="InvalidParameterValue"
message="ia:InvalidParameterValueFault" />
        <wsdl:fault name="MissingParameterValue"
message="ia:MissingParameterValueFault" />
        <wsdl:fault name="NoApplicableCode"
message="ia:NoApplicableCodeFault" />
        <wsdl:fault name="InternalError"
message="ia:InternalErrorFault" />
        <wsdl:fault name="PermissionDenied"
message="ia:PermissionDeniedFault" />
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="IdentityManagementAndAuthenticationService"
type="ia:IdentityManagementAndAuthenticationService">
    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="getCapabilities">
        <soap:operation soapAction="getCapabilities" style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
        <wsdl:fault name="InvalidParameterValue">
            <soap:fault name="InvalidParameterValue" use="literal" />
        </wsdl:fault>
        <wsdl:fault name="MissingParameterValue">
            <soap:fault name="MissingParameterValue" use="literal" />
        </wsdl:fault>
        <wsdl:fault name="NoApplicableCode">
            <soap:fault name="NoApplicableCode" use="literal" />
        </wsdl:fault>
        <wsdl:fault name="InternalError">
            <soap:fault name="InternalError" use="literal" />
        </wsdl:fault>
        <wsdl:fault name="UnsupportedCapSchema">
            <soap:fault name="UnsupportedCapSchema" use="literal" />
        </wsdl:fault>
        <wsdl:fault name="VersionNegotiationFailed">
            <soap:fault name="VersionNegotiationFailed" use="literal" />
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="verifySessionInformation">
        <wsdl:documentation>The verifySessionInformation operation
determines whether the given session-information is
valid.</wsdl:documentation>
        <soap:operation soapAction="verifySessionInformation"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>

```

```

<wsdl:operation name="updateIdentity">
  <soap:operation soapAction="updateIdentity" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalError">
    <soap:fault name="InternalError" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="IdentityNotFound">
    <soap:fault name="IdentityNotFound" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="PermissionDenied">
    <soap:fault name="PermissionDenied" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="updateCredentials">
  <soap:operation soapAction="updateCredentials" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalError">
    <soap:fault name="InternalError" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="PermissionDenied">
    <soap:fault name="PermissionDenied" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="deleteIdentity">
  <soap:operation soapAction="deleteIdentity" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalError">
    <soap:fault name="InternalError" use="literal"/>
  </wsdl:fault>

```

```

        <soap:fault name="InternalError" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="IdentityNotFound">
        <soap:fault name="IdentityNotFound" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="PermissionDenied">
        <soap:fault name="PermissionDenied" use="literal"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="deleteCredentials">
    <soap:operation soapAction="deleteCredentials" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:fault name="InvalidParameterValue">
        <soap:fault name="InvalidParameterValue" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="MissingParameterValue">
        <soap:fault name="MissingParameterValue" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="NoApplicableCode">
        <soap:fault name="NoApplicableCode" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="InternalError">
        <soap:fault name="InternalError" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="IdentityNotFound">
        <soap:fault name="IdentityNotFound" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="PermissionDenied">
        <soap:fault name="PermissionDenied" use="literal"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="login">
    <soap:operation soapAction="login" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="deactivateIdentity">
    <soap:operation soapAction="deactivateIdentity"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:fault name="InvalidParameterValue">
        <soap:fault name="InvalidParameterValue" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="MissingParameterValue">
        <soap:fault name="MissingParameterValue" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="NoApplicableCode">
        <soap:fault name="NoApplicableCode" use="literal"/>
    </wsdl:fault>
    <wsdl:fault name="InternalError">
        <soap:fault name="InternalError" use="literal"/>
    </wsdl:fault>

```

```

<wsdl:fault name="IdentityNotFound">
  <soap:fault name="IdentityNotFound" use="literal"/>
</wsdl:fault>
<wsdl:fault name="PermissionDenied">
  <soap:fault name="PermissionDenied" use="literal"/>
</wsdl:fault>
</wsdl:operation>
<wsdl:operation name="createIdentity">
  <soap:operation soapAction="createIdentity" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalServerError">
    <soap:fault name="InternalServerError" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="PermissionDenied">
    <soap:fault name="PermissionDenied" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="addCredentials">
  <soap:operation soapAction="addCredentials" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalServerError">
    <soap:fault name="InternalServerError" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="IdentityNotFound">
    <soap:fault name="IdentityNotFound" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="PermissionDenied">
    <soap:fault name="PermissionDenied" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="activateIdentity">
  <soap:operation soapAction="activateIdentity" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>

```

```

<wsdl:fault name="MissingParameterValue">
  <soap:fault name="MissingParameterValue" use="literal"/>
</wsdl:fault>
<wsdl:fault name="NoApplicableCode">
  <soap:fault name="NoApplicableCode" use="literal"/>
</wsdl:fault>
<wsdl:fault name="InternalError">
  <soap:fault name="InternalError" use="literal"/>
</wsdl:fault>
<wsdl:fault name="IdentityNotFound">
  <soap:fault name="IdentityNotFound" use="literal"/>
</wsdl:fault>
<wsdl:fault name="PermissionDenied">
  <soap:fault name="PermissionDenied" use="literal"/>
</wsdl:fault>
</wsdl:operation>
<wsdl:operation name="getIdentities">
  <soap:operation soapAction="getIdentities" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="InvalidParameterValue">
    <soap:fault name="InvalidParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="MissingParameterValue">
    <soap:fault name="MissingParameterValue" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="NoApplicableCode">
    <soap:fault name="NoApplicableCode" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="InternalError">
    <soap:fault name="InternalError" use="literal"/>
  </wsdl:fault>
  <wsdl:fault name="PermissionDenied">
    <soap:fault name="PermissionDenied" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="IdentityManagementAndAuthenticationService">
  <wsdl:port name="IdentityManagementAndAuthenticationService"
binding="ia:IdentityManagementAndAuthenticationService">
    <soap:address
location="http://localhost:8080/axis2/services/IdentityManagementAndAuthentic
ationService"/>
  </wsdl:port>
</wsdl:service>
<plnk:partnerLinkType name="IdentityManagementAndAuthenticationService">
  <plnk:role name="role1"
portType="ia:IdentityManagementAndAuthenticationService"/>
</plnk:partnerLinkType>
</wsdl:definitions>

```

8.3. Capabilities Document Template

The following XML document defines a template for the capabilities document of the service:

```

<oami:OA_MI_Service_Capabilities
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://eu-orchestra.org/OAS-MI/service/1.1
http://www.enviomatics.net/WS/OrchestraArchive/oasmi.xsd
http://eu-orchestra.org/OA/OABasicService/types/1.0
http://www.enviomatics.net/WS/OrchestraArchive/oa_basic.xsd

http://www.enviomatics.net/WS/IdentityManagementAndAuthenticationService/mi/
2.0
http://www.enviomatics.net/WS/IdentityManagementAndAuthenticationService/mi/
2.0/identity_authen_mi.xsd
http://eu-orchestra.org/OAS-MI/service/invocation/1.1
inv.xsd"
  xmlns:oami="http://eu-orchestra.org/OAS-MI/service/1.1"
  xmlns:oabs="http://eu-orchestra.org/OA/OABasicService/types/1.0"

  xmlns:imasoami="http://www.enviomatics.net/WS/IdentityManagementAndAuthentic
ationService/mi/2.0"
  xmlns:inv="http://eu-orchestra.org/OAS-MI/service/invocation/1.1">

  <oami:serviceCommonCapabilities>
    <oami:OA_MI_Service_CommonCapabilities>
      <oami:id>Identity Management and Authentication Service
1.0</oami:id>
      <oami:publicationDate>2009-01-13</oami:publicationDate>
      <oami:serviceDescription>
        Identities are managed (created, deleted, etc.) using an
Identity Management Interface instance. The Identity Management Interface
acts as an identity provider (IdP).
        The Authentication Interface verifies genuineness of
identities using a set of given credentials.
      </oami:serviceDescription>
      <oami:serviceDocumentation>http://sany-
ip.eu/</oami:serviceDocumentation>
      <oami:serviceInvocationBasic>
        <inv:OA_MI_Service_InvocationBasic>
          <inv:operation>
            <inv:OA_MI_Operation>
              <inv:accessPoints>
                <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthentica
tionService</inv:uri>

                </inv:OA_MI_AccessPoint>
              </inv:accessPoints>
              <inv:description>The mandatory getCapabilities
operation informs the client of the capabilities of an service instance. This
operation takes into account that in addition to capabilities that may be
common to all services in a service network a service may provide a specific
set of capabilities.</inv:description>
              <inv:name>getCapabilities</inv:name>
              <inv:parameters>
                <inv:OA_MI_OperationParameter>

```

```

        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>OA_GetCapabilitiesRequest</inv:valueType>
    </inv:OA_MI_OperationParameter>
</inv:parameters>
<inv:parameters>
    <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>out</inv:direction>
        <inv:name>reponse</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>OA_GetCapabilitiesResponse</inv:valueType>
    </inv:OA_MI_OperationParameter>
</inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
    <inv:OA_MI_Operation>
        <inv:accessPoints>
            <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
            </inv:OA_MI_AccessPoint>
        </inv:accessPoints>
        <inv:description>
            The Mandatory login operation initiates
            the validation of a certain identity for given credential. It returns session
            information in form of a SAML 2.0 token which contains assertions for the
            authenticated identity as well as assertions for the group identities to
            which this identity is associated.
        </inv:description>
        <inv:name>login</inv:name>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>in</inv:direction>
                <inv:name>request</inv:name>
                <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>AuthnRequest</inv:valueType>
    </inv:OA_MI_OperationParameter>
</inv:parameters>
<inv:parameters>
    <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>out</inv:direction>
        <inv:name>response</inv:name>
        <inv:optionality>>false</inv:optionality>

```

```

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>ResponseType</inv:valueType>
  </inv:OA_MI_OperationParameter>
  </inv:parameters>
  </inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>
      The mandatory activate Identity operation
      activates an existing identity. Only active identities can be authenticated.
    </inv:description>
    <inv:name>activatePrincipal</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>activateIdentityRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>
  </inv:parameters>
  </inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>
      The mandatory addCredentials operation
      adds credentials to a certain identity. Credentials are specific to the
      authentication mechanism used. The current specifications support username /
      password credentials that can be added to UsernameIdentities.
    </inv:description>
    <inv:name>addCredentials</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

```

```

<inv:valueType>addCredentialsRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>
</inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>The mandatory createPrincipal
operation creates a new principal. The parameter is a principal repre-
sentation that is specific to the used authentication mechanism.
For a username/password authentication the principal contains at least a
username.

    </inv:description>
    <inv:name>createPrincipal</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>createPrincipalRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>
</inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>The optional deactivatePrincipal
operation deactivates an existing principal. Only active principals can be
authenticated.</inv:description>
    <inv:name>deactivatePrincipal</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>principal</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>deactivatePrincipalRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>

```

```

        </inv:parameters>
    </inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
    <inv:OA_MI_Operation>
        <inv:accessPoints>
            <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthentica
tionService</inv:uri>
            </inv:OA_MI_AccessPoint>
        </inv:accessPoints>
        <inv:description>The mandatory getPrincipals
operation allows the retrieval of a list of principals specified by a given
query.</inv:description>
        <inv:name>getPrincipals</inv:name>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>in</inv:direction>
                <inv:name>request</inv:name>
                <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>getPrincipalsRequest</inv:valueType>
            </inv:OA_MI_OperationParameter>
        </inv:parameters>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>out</inv:direction>
                <inv:name>response</inv:name>
                <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>getPrincipalsResponse</inv:valueType>
            </inv:OA_MI_OperationParameter>
        </inv:parameters>
    </inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
    <inv:OA_MI_Operation>
        <inv:accessPoints>
            <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthentica
tionService</inv:uri>
            </inv:OA_MI_AccessPoint>
        </inv:accessPoints>
        <inv:description>The optional removeCredentials
operation removes credentials from a given principal.</inv:description>
        <inv:name>removeCredentials</inv:name>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>in</inv:direction>
                <inv:name>request</inv:name>
                <inv:optionality>>false</inv:optionality>

```

```

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>removeCredentialsRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>
</inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>The mandatory removePrincipal
operation creates deletes an existing principal. The principal represen-
tation is specific to the authentication mechanism used. To keep consistency
the deletion of a principal should be performed within a transaction. In this
transaction all references to the principal should also get deleted (e.g.
group memberships).</inv:description>
    <inv:name>removePrincipal</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>removePrincipalRequest</inv:valueType>
  </inv:OA_MI_OperationParameter>
</inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
  <inv:OA_MI_Operation>
    <inv:accessPoints>
      <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
      </inv:OA_MI_AccessPoint>
    </inv:accessPoints>
    <inv:description>The mandatory updateCredentials
operation updates credentials for a certain principal.</inv:description>
    <inv:name>updateCredentials</inv:name>
    <inv:parameters>
      <inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>in</inv:direction>
        <inv:name>request</inv:name>
        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>updateCredentialsRequest</inv:valueType>

```

```

        </inv:OA_MI_OperationParameter>
    </inv:parameters>
</inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
    <inv:OA_MI_Operation>
        <inv:accessPoints>
            <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthentica
tionService</inv:uri>

            </inv:OA_MI_AccessPoint>
        </inv:accessPoints>
        <inv:description>The mandatory updatePrincipal
operation updates an existing principal.</inv:description>
        <inv:name>updatePrincipal</inv:name>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>in</inv:direction>
                <inv:name>request</inv:name>
                <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>updatePrincipalRequest</inv:valueType>
            </inv:OA_MI_OperationParameter>
        </inv:parameters>
    </inv:OA_MI_Operation>
</inv:operation>
<inv:operation>
    <inv:OA_MI_Operation>
        <inv:accessPoints>
            <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthentica
tionService</inv:uri>

            </inv:OA_MI_AccessPoint>
        </inv:accessPoints>
        <inv:description>The verifySessionInformation
operation determines whether the given session-information is
valid.</inv:description>
        <inv:name>verifySessionInformation</inv:name>
        <inv:parameters>
            <inv:OA_MI_OperationParameter>
                <inv:description></inv:description>
                <inv:direction>in</inv:direction>
                <inv:name>request</inv:name>
                <inv:optionality>>false</inv:optionality>

<inv:repeatability>true</inv:repeatability>

<inv:valueType>verifySessionInformationRequest</inv:valueType>
            </inv:OA_MI_OperationParameter>
        </inv:parameters>
    </inv:OA_MI_OperationParameter>
        <inv:description></inv:description>
        <inv:direction>out</inv:direction>
        <inv:name>response</inv:name>

```

```

        <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>verifySessionInformationResponse</inv:valueType>
    </inv:OA_MI_OperationParameter>
    </inv:parameters>
    </inv:OA_MI_Operation>
  </inv:operation>
</inv:OA_MI_Service_InvocationBasic>
</oami:serviceInvocationBasic>
<oami:serviceName>Authentication Service</oami:serviceName>
<oami:serviceSpecVersion>
  0.9.1
</oami:serviceSpecVersion>
<oami:serviceType>
  <oabs:identifier>Authentication Service</oabs:identifier>
</oami:serviceType>
</oami:OA_MI_Service_CommonCapabilities>
</oami:serviceCommonCapabilities>
<oami:serviceSpecificCapabilities>
  <imasoami:MI_AuthenticationServiceCapabilities>
    <imasoami:supportedAuthenticationMechanisms>
      <imasoami:MI_AuthenticationMechanism>
        <inv:operation>
          <inv:OA_MI_Operation>
            <inv:accessPoints>
              <inv:OA_MI_AccessPoint>

<inv:uri>http://localhost:8080/axis2/services/IdentityManagementandAuthenticati
tionService</inv:uri>
              </inv:OA_MI_AccessPoint>
            </inv:accessPoints>
          <inv:description>
            The Mandatory login operation initiates
            the validation of a certain identity for given credential. It returns session
            information in form of a SAML 2.0 token which contains assertions for the
            authenticated identity as well as assertions for the group identities to
            which this identity is associated.
          </inv:description>
          <inv:name>login</inv:name>
        <inv:parameters>
          <inv:OA_MI_OperationParameter>
            <inv:description></inv:description>
            <inv:direction>in</inv:direction>
            <inv:name>request</inv:name>
            <inv:optionality>>false</inv:optionality>

<inv:repeatability>>true</inv:repeatability>

<inv:valueType>AuthnRequest</inv:valueType>
    </inv:OA_MI_OperationParameter>
  </inv:parameters>
<inv:parameters>
  <inv:OA_MI_OperationParameter>
    <inv:description></inv:description>
    <inv:direction>out</inv:direction>
    <inv:name>response</inv:name>
    <inv:optionality>>false</inv:optionality>

```

```
<inv:repeatability>true</inv:repeatability>

<inv:valueType>ResponseType</inv:valueType>
  </inv:OA_MI_OperationParameter>
  </inv:parameters>
  </inv:OA_MI_Operation>
</inv:operation>
  <imasoami:name>UserPassword</imasoami:name>
  </imasoami:MI_AuthenticationMechanism>
  </imasoami:supportedAuthenticationMechanisms>
  </imasoami:MI_AuthenticationServiceCapabilities>
  </oami:serviceSpecificCapabilities>
</oami:OA_MI_Service_Capabilities>
```