



User Authentication and Authorization

Table of Contents

Table of Contents	1
1. INTRODUCTION	3
2. OBJECTIVES	3
3. SCOPE	3
4. PROFILE DETAILED DESCRIPTION	4
4.2. Roles Involved	4
4.3. Actor Diagram	4
4.4. Transaction Information	5
4.4.1. Authenticate User with Credentials	5
4.4.1.1. Description	5
4.4.1.2. Interaction Diagram	5
4.4.1.3. Authenticate User with Credentials Message	6
4.4.1.3.1. Message Semantics	6
4.4.1.3.2. Protocol Requirements	7
4.4.1.4. Send Token Message	7
4.4.1.4.1. Message Semantics	7
4.4.1.4.2. Protocol Requirements	7
4.4.2. Authenticate User with Token	7
4.4.2.1. Description	7
4.4.2.2. Interaction Diagram	8
4.4.2.3. Authenticate User with Token Message	8
4.4.2.3.1. Message Semantics	8
4.4.2.3.2. Protocol Requirements	9
4.4.2.4. Validate Token Message	9
4.4.2.4.1. Message Semantics	9
4.4.2.4.2. Protocol Requirements	9
4.4.3. Authorize User	10
4.4.3.1. Description	10
4.4.3.2. Interaction Diagram	10

4.4.3.3. Grant User Permission Message	10
4.4.3.3.1. Message Semantics.....	10
4.4.3.3.2. Protocol Requirements	12
4.4.4. Check Authorization.....	12
4.4.4.1. Description	12
4.4.4.2. Interaction Diagram	13
4.4.4.3. Check Authorization Message.....	13
4.4.4.3.1. Message Semantics.....	13
4.4.4.3.2. Protocol Requirements	15
4.4.3.4. Respond to Check Authorization Message	15
4.4.3.4.1. Message Semantics.....	15
4.4.3.4.2. Protocol Requirements	17

1. INTRODUCTION

User Authentication and Authorization defines a means to establish one name per user that can then be used on all of the devices and software that participate in this integration profile. It greatly facilitates decentralized user authentication management and provides users with the convenience and speed of a single sign-on.

2. OBJECTIVES

The objectives of implementing this profile are:

1. Verifying the identity of a user, process, or device.
2. Facilitating authentication operation through a single sign-on mechanism.
3. Ensuring that the resources are accessible only to those people who are authorized to do so.

3. SCOPE

This profile supports a process of authentication and authorization of users. It is intended to identify users as well as specifying access rights of them. The intended scope for this profile includes:

- Before, during and after emergency situations (where emergency systems are in use)
- In warning systems, emergency information systems, sensor systems, dissemination systems and internal systems.

The transactions specified in this profile are intended to be exchanged between emergency organizations' systems and authentication and authorization servers. It is expected that emergency systems have had necessary infrastructure, which is specified in the profile, to send and receive the messages in a secured way, using an agreed syntax.

4. PROFILE DETAILED DESCRIPTION

4.2. Roles Involved

Actor	Description
Application Client	Any application that requires authentication or authorization of the user who wants to use it
Authentication Server	Server which is responsible from authentication operations
Authorization Server	Server which is responsible from authorization operations

4.3. Actor Diagram

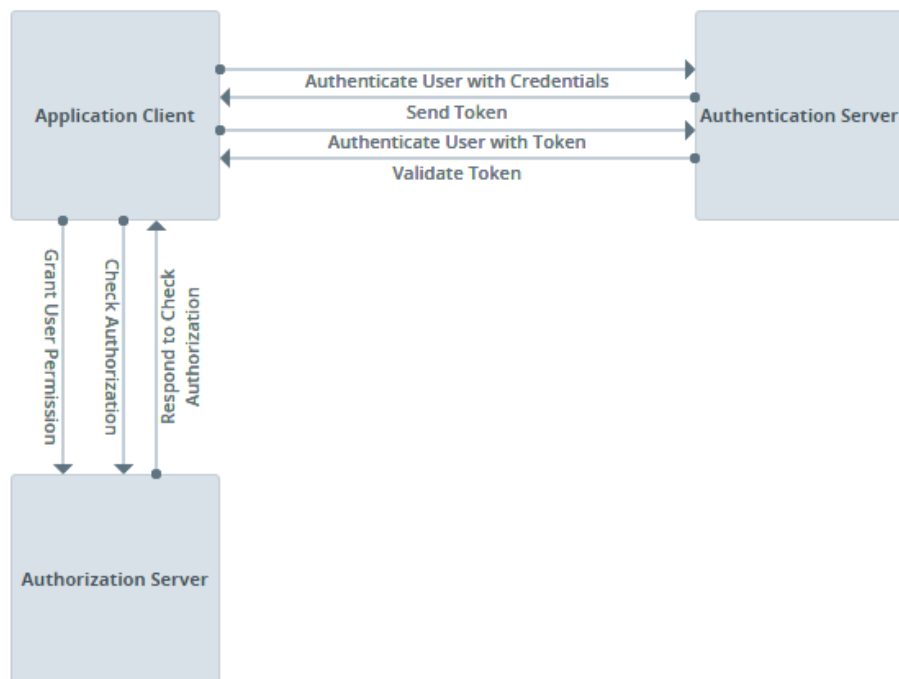


Figure above shows the actors directly involved in the User Authentication and Authorization Profile and the possible messages to be exchanged between them.

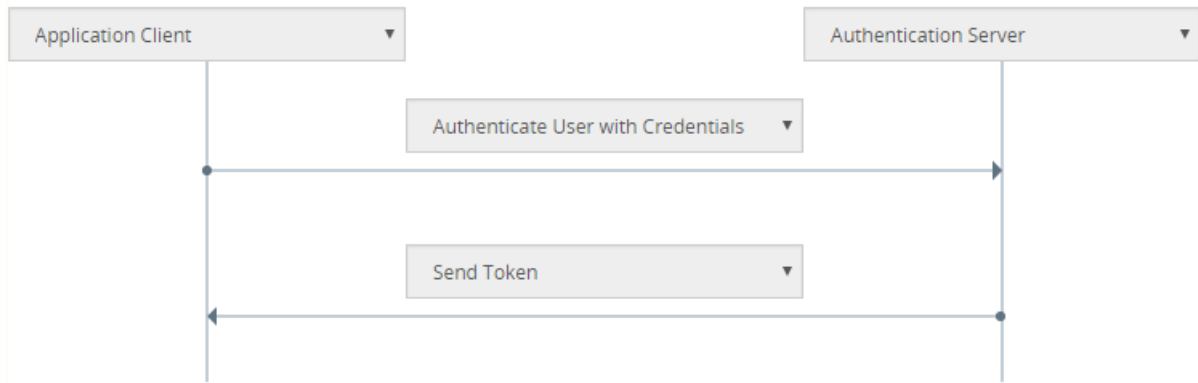
4.4. Transaction Information

4.4.1. Authenticate User with Credentials

4.4.1.1. Description

Categories	Description and Values
Identifier	1461504247128
Description	This transaction is used to authenticate user identity with username/password. A challenge-response method verifies that the user knows the correct password. Once the user is authenticated, the Authentication Server sends an Authentication Token to the Application Client. The Authentication Token acts as a substitute for repeated login/password type activity.
Actors	Application Client, Authentication Server
Pre-conditions	User does not have a token already (It might be a new user or user might have been logged out or ticket might have expired)
Post-conditions	<p>If login is successful, user is authenticated and provided a token.</p> <p>If login is unsuccessful, user is informed that authentication failed.</p>

4.4.1.2. Interaction Diagram



4.4.1.3. Authenticate User with Credentials Message

4.4.1.3.1. Message Semantics

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns:Q1="http://www.c2sense.eu/AuthenticationRequest">
  <element name="AuthenticationRequest" type="Q1:AuthenticationRequestType"></element>
  <complexType name="AuthenticationRequestType">
    <sequence>
      <element name="credential" minOccurs="0" maxOccurs="1">
        <complexType>
          <sequence>
            <element name="username" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <element name="password" type="xs:string" minOccurs="1" maxOccurs="1"/>
          </sequence>
        </complexType>
      </element>
      <element name="token" minOccurs="0" maxOccurs="1">
        <complexType>
          <sequence>
            <element name="access_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
  </complexType>
</xs:schema>
  
```

4.4.1.3.2. Protocol Requirements

Use REST protocol and handle it through POST request.

4.4.1.4. Send Token Message

4.4.1.4.1. Message Semantics

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns:Q1="http://www.c2sense.eu/AuthenticationResponse">
  <element name="AuthenticationResponse" type="Q1:AuthenticationResponseType"></element>
  <complexType name="AuthenticationResponseType">
    <sequence>
      <element name="access_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="token_type" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="refresh_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="expires_in" type="xs:integer" minOccurs="1" maxOccurs="1"/>
      <element name="id_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
    </sequence>
  </complexType>
</xs:schema>
```

4.4.1.4.2. Protocol Requirements

Use REST protocol and handle it through POST request.

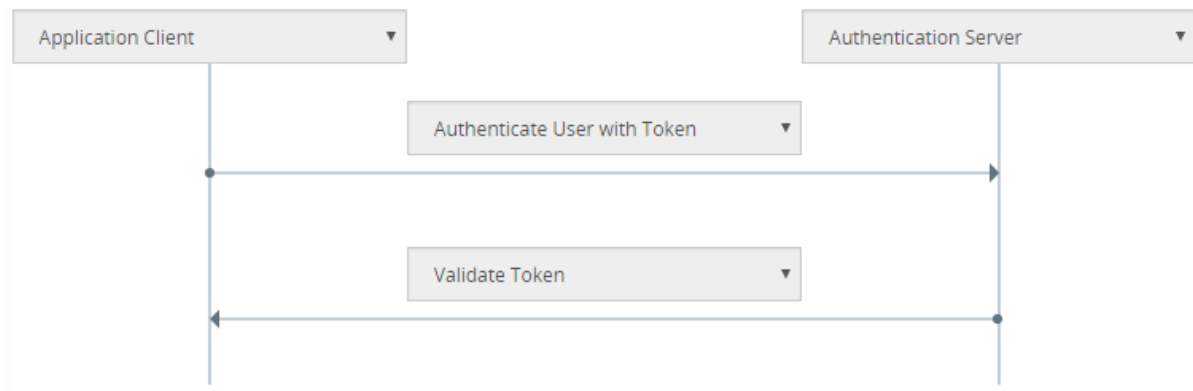
4.4.2. Authenticate User with Token

4.4.2.1. Description

Categories	Description and Values
Identifier	1461504247311
Description	This transaction is used to authenticate user identity with Authentication Token. If the token is valid, user is authenticated.

Actors	Application Client, Authentication Server
Pre-conditions	User has a token already.
Post-conditions	<p>If token is valid, user is authenticated.</p> <p>If token is invalid, user is informed that authentication failed.</p>

4.4.2.2. Interaction Diagram



4.4.2.3. Authenticate User with Token Message

4.4.2.3.1. Message Semantics

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns:Q1="http://www.c2sense.eu/AuthenticationRequest">
  <element name="AuthenticationRequest" type="Q1:AuthenticationRequestType"></element>
  <complexType name="AuthenticationRequestType">
    <sequence>
      <element name="credential" minOccurs="0" maxOccurs="1">
        <complexType>
          <sequence>
            <element name="username" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <element name="password" type="xs:string" minOccurs="1" maxOccurs="1"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
  </complexType>

```

```
</element>
<element name="token" minOccurs="0" maxOccurs="1">
  <complexType>
    <sequence>
      <element name="access_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
    </sequence>
  </complexType>
</element>
</sequence>
</complexType>
</xs:schema>
```

4.4.2.3.2. Protocol Requirements

Use REST protocol and handle it through POST request.

4.4.2.4. Validate Token Message

4.4.2.4.1. Message Semantics

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns:Q1="http://www.c2sense.eu/AuthenticationResponse">
  <element name="AuthenticationResponse" type="Q1:AuthenticationResponseType"></element>
  <complexType name="AuthenticationResponseType">
    <sequence>
      <element name="access_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="token_type" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="refresh_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <element name="expires_in" type="xs:integer" minOccurs="1" maxOccurs="1"/>
      <element name="id_token" type="xs:string" minOccurs="1" maxOccurs="1"/>
    </sequence>
  </complexType>
</xs:schema>
```

4.4.2.4.2. Protocol Requirements

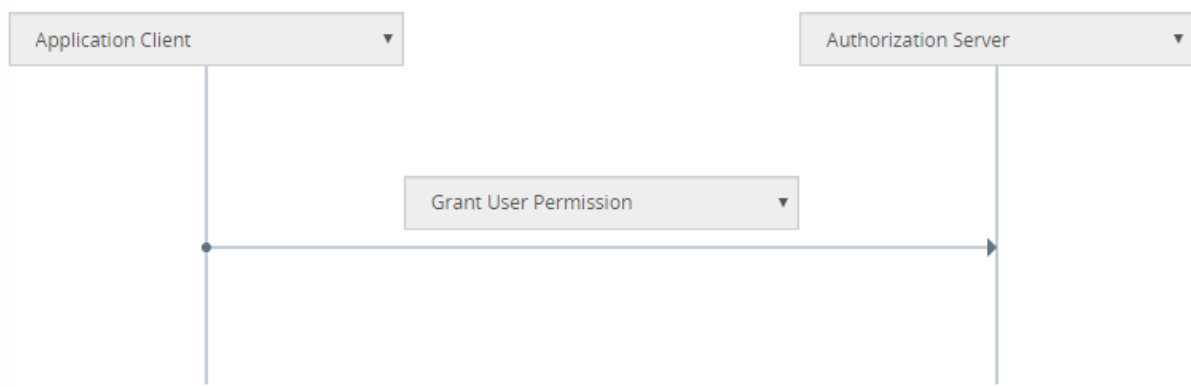
Use REST protocol and handle it through POST request.

4.4.3. Authorize User

4.4.3.1. Description

Categories	Description and Values
Identifier	24f595dc-4c54-42e8-b2a2-7732722c3272
Description	This transaction is used to grant a user permission to perform a specific action.
Actors	Application Client, Authorization Server
Pre-conditions	User is already defined in the system. Action is already defined in the system.
Post-conditions	User is granted permission to perform a specific action.

4.4.3.2. Interaction Diagram



4.4.3.3. Grant User Permission Message

4.4.3.3.1. Message Semantics

```

<?xml version="1.0" encoding="UTF-8"?>
<schema
  targetNamespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"

```

```
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:xacml="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
xmlns:xacml-context="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
xmlns:xacml-saml="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01"
xmlns:xacml-samlp="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"
elementFormDefault="unqualified"
attributeFormDefault="unqualified"
blockDefault="substitution"
version="CD 1">
  <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="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11" schemaLocation="xacml-
core-v3-schema-wd-11.xsd"/>
  <import namespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01"
schemaLocation="xacml-3.0-profile-saml2.0-v2-schema-assertion-cd-1.xsd" />
  <element name="XACMLAuthzDecisionQuery" xsi:type="xacml-samlp:XACMLAuthzDecisionQueryType" />
  <complexType name="XACMLAuthzDecisionQueryType">
    <complexContent>
      <extension base="samlp:RequestAbstractType">
        <sequence>
          <element ref="xacml-context:Request" />
          <element ref="xacml-samlp:AdditionalAttributes" minOccurs="0" maxOccurs="1"
/>
          <element ref="xacml:Policy" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml:PolicySet" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml-saml:ReferencedPolicies" minOccurs="0" maxOccurs="1" />
        </sequence>
        <attribute name="InputContextOnly" type="boolean" use="optional" default="false"
/>
        <attribute name="ReturnContext" type="boolean" use="optional" default="false" />
        <attribute name="CombinePolicies" type="boolean" use="optional" default="true" />
      </extension>
    </complexContent>
  </complexType>
  <element name="AdditionalAttributes" xsi:type="xacml-samlp:AdditionalAttributesType" />
  <complexType name="AdditionalAttributesType">
    <sequence>
      <element ref="xacml-samlp:AssignedAttributes" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <element name="AssignedAttributes" xsi:type="xacml-samlp:AssignedAttributesType" />
  <complexType name="AssignedAttributesType">
    <sequence>
      <element ref="xacml-samlp:Holders" />
      <element ref="xacml-samlp:HolderAttributes" />
    </sequence>
  </complexType>
  <element name="Holders" xsi:type="xacml-samlp:HoldersType" />
  <complexType name="HoldersType">
```

```
<sequence>
  <element ref="xacml:Match" maxOccurs="unbounded" />
</sequence>
</complexType>
<element name="HolderAttributes" xsi:type="xacml-samlp:HolderAttributesType" />
<complexType name="HolderAttributesType">
  <sequence>
    <element ref="xacml-context:Attribute" minOccurs="0" maxOccurs="unbounded" />
  </sequence>
</complexType>
<element name="XACMLPolicyQuery" xsi:type="xacml-samlp:XACMLPolicyQueryType" />
<complexType name="XACMLPolicyQueryType">
  <complexContent>
    <extension base="samlp:RequestAbstractType">
      <choice minOccurs="1" maxOccurs="unbounded">
        <element ref="xacml-context:Request" />
        <element ref="xacml:PolicySetIdReference" />
        <element ref="xacml:PolicyIdReference" />
      </choice>
    </extension>
  </complexContent>
</complexType>
</schema>
```

4.4.3.3.2. Protocol Requirements

Use REST protocol and handle it through POST request.

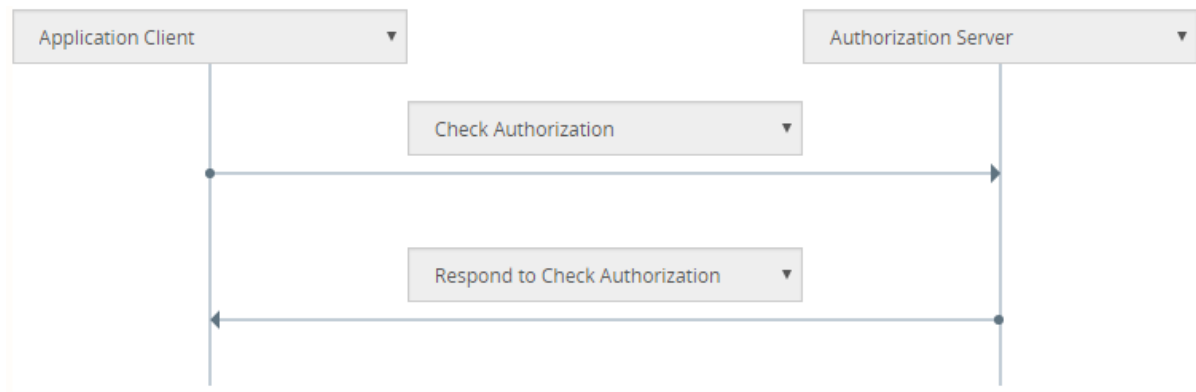
4.4.4. Check Authorization

4.4.4.1. Description

Categories	Description and Values
Identifier	30a1408d-8093-4e53-ac7a-8b6ea8c0e17e
Description	This transaction is used to checker whether user is already granted permission to perform a specific action.
Actors	Application Client, Authorization Server

Pre-conditions	<p>User is already defined in the system.</p> <p>Action is already defined in the system.</p> <p>User is requested to perform the specific action.</p>
Post-conditions	<p>If user is authorized, he is allowed to perform the action.</p> <p>If user is not authorized, he is rejected to perform the action.</p>

4.4.4.2. Interaction Diagram



4.4.4.3. Check Authorization Message

4.4.4.3.1. Message Semantics

```

<?xml version="1.0" encoding="UTF-8"?>
<schema
  targetNamespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xacml="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
  xmlns:xacml-context="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
  xmlns:xacml-saml="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01"
  xmlns:xacml-samlp="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"
  elementFormDefault="unqualified"
  attributeFormDefault="unqualified"
  blockDefault="substitution"

```

```
version="CD 1">
  <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="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11" schemaLocation="xacml-
core-v3-schema-wd-11.xsd"/>
  <import namespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01"
schemaLocation="xacml-3.0-profile-saml2.0-v2-schema-assertion-cd-1.xsd" />
  <element name="XACMLAuthzDecisionQuery" xsi:type="xacml-samlp:XACMLAuthzDecisionQueryType" />
  <complexType name="XACMLAuthzDecisionQueryType">
    <complexContent>
      <extension base="samlp:RequestAbstractType">
        <sequence>
          <element ref="xacml-context:Request" />
          <element ref="xacml-samlp:AdditionalAttributes" minOccurs="0" maxOccurs="1"
/>

          <element ref="xacml:Policy" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml:PolicySet" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml-saml:ReferencedPolicies" minOccurs="0" maxOccurs="1" />
        </sequence>
        <attribute name="InputContextOnly" type="boolean" use="optional" default="false"
/>

        <attribute name="ReturnContext" type="boolean" use="optional" default="false" />
        <attribute name="CombinePolicies" type="boolean" use="optional" default="true" />
      </extension>
    </complexContent>
  </complexType>
  <element name="AdditionalAttributes" xsi:type="xacml-samlp:AdditionalAttributesType" />
  <complexType name="AdditionalAttributesType">
    <sequence>
      <element ref="xacml-samlp:AssignedAttributes" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <element name="AssignedAttributes" xsi:type="xacml-samlp:AssignedAttributesType" />
  <complexType name="AssignedAttributesType">
    <sequence>
      <element ref="xacml-samlp:Holders" />
      <element ref="xacml-samlp:HolderAttributes" />
    </sequence>
  </complexType>
  <element name="Holders" xsi:type="xacml-samlp:HoldersType" />
  <complexType name="HoldersType">
    <sequence>
      <element ref="xacml:Match" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <element name="HolderAttributes" xsi:type="xacml-samlp:HolderAttributesType" />
  <complexType name="HolderAttributesType">
    <sequence>
      <element ref="xacml-context:Attribute" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
  </complexType>
```

```
<element name="XACMLPolicyQuery" xsi:type="xacml-samlp:XACMLPolicyQueryType" />
<complexType name="XACMLPolicyQueryType">
  <complexContent>
    <extension base="samlp:RequestAbstractType">
      <choice minOccurs="1" maxOccurs="unbounded">
        <element ref="xacml-context:Request" />
        <element ref="xacml:PolicySetIdReference" />
        <element ref="xacml:PolicyIdReference" />
      </choice>
    </extension>
  </complexContent>
</complexType>
</schema>
```

4.4.4.3.2. Protocol Requirements

Use REST protocol and handle it through POST request.

4.4.3.4. Respond to Check Authorization Message

4.4.3.4.1. Message Semantics

```
<?xml version="1.0" encoding="UTF-8"?>
<schema
  targetNamespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xacml="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
  xmlns:xacml-context="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11"
  xmlns:xacml-saml="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01"
  xmlns:xacml-samlp="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:protocol:cd-01"
  elementFormDefault="unqualified"
  attributeFormDefault="unqualified"
  blockDefault="substitution"
  version="CD 1">
  <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="urn:oasis:names:tc:xacml:3.0:core:schema:wd-11" schemaLocation="xacml-core-v3-schema-wd-11.xsd"/>
  <import namespace="urn:oasis:names:tc:xacml:3.0:profile:saml2.0:v2:schema:assertion:cd-01" schemaLocation="xacml-3.0-profile-saml2.0-v2-schema-assertion-cd-1.xsd" />
  <element name="XACMLAuthzDecisionQuery" xsi:type="xacml-samlp:XACMLAuthzDecisionQueryType" />
  <complexType name="XACMLAuthzDecisionQueryType">
```



```

    <complexContent>
      <extension base="samlp:RequestAbstractType">
        <sequence>
          <element ref="xacml-context:Request" />
          <element ref="xacml-samlp:AdditionalAttributes" minOccurs="0" maxOccurs="1"
/>
          <element ref="xacml:Policy" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml:PolicySet" minOccurs="0" maxOccurs="unbounded" />
          <element ref="xacml-saml:ReferencedPolicies" minOccurs="0" maxOccurs="1" />
        </sequence>
        <attribute name="InputContextOnly" type="boolean" use="optional" default="false"
/>
        <attribute name="ReturnContext" type="boolean" use="optional" default="false" />
        <attribute name="CombinePolicies" type="boolean" use="optional" default="true" />
      </extension>
    </complexContent>
  </complexType>
  <element name="AdditionalAttributes" xsi:type="xacml-samlp:AdditionalAttributesType" />
  <complexType name="AdditionalAttributesType">
    <sequence>
      <element ref="xacml-samlp:AssignedAttributes" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <element name="AssignedAttributes" xsi:type="xacml-samlp:AssignedAttributesType" />
  <complexType name="AssignedAttributesType">
    <sequence>
      <element ref="xacml-samlp:Holders" />
      <element ref="xacml-samlp:HolderAttributes" />
    </sequence>
  </complexType>
  <element name="Holders" xsi:type="xacml-samlp:HoldersType" />
  <complexType name="HoldersType">
    <sequence>
      <element ref="xacml:Match" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <element name="HolderAttributes" xsi:type="xacml-samlp:HolderAttributesType" />
  <complexType name="HolderAttributesType">
    <sequence>
      <element ref="xacml-context:Attribute" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <element name="XACMLPolicyQuery" xsi:type="xacml-samlp:XACMLPolicyQueryType" />
  <complexType name="XACMLPolicyQueryType">
    <complexContent>
      <extension base="samlp:RequestAbstractType">
        <choice minOccurs="1" maxOccurs="unbounded">
          <element ref="xacml-context:Request" />
          <element ref="xacml:PolicySetIdReference" />
          <element ref="xacml:PolicyIdReference" />

```



```
        </choice>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

4.4.3.4.2. Protocol Requirements

Use REST protocol and handle it through POST request.