Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Dipartimento federale dell'interno DFI

RS 816.111
Supplemento 2 all'allegato 5 dell'ordinanza del DFI del 22 marzo 2017 sulla cartella informatizzata del paziente

Profili d'integrazione nazionale secondo l'articolo 5 capoverso 1 lettera c OCIP-DFI

Authorization Decision Request (CH:ADR) and Privacy Policy Query (CH:PPQ)

Edizione 1: 22 marzo 2017
Entrata in vigore: 15 aprile 2017

1 Introduction

La cartella informatizzata del paziente (CIP) si basa su un sistema che prevede numerose comunità IHE XDS, in cui il paziente non accorda solo il consenso per la costituzione e l'utilizzo della sua cartella, ma stabilisce esplicitamente anche regole per l'accesso tramite un apposito portale per pazienti.

Le impostazioni della protezione dei dati del paziente (per quanto concerne l'accesso alla sua cartella informatizzata) è memorizzata dalla comunità di riferimento e tutti i sistemi coinvolti devono attenervisi. È stato specificato che gli elenchi dei documenti fungono da Policy Enforcing Service Provider in forma di un XACML PEP (Policy Enforcement Point). Tuttavia, siccome le regole da far rispettare non sono a disposizione nell'elenco dei documenti di una comunità, il XACML-PDP (Policy Decision Point) deve essere implementato come attore a sé stante, per assicurare l'interoperabilità per quanto concerne le direttive da far rispettare. Inoltre gli stessi Policy Repositories (XACML PAP) devono fungere da Policy Enforcing Service Provider.

La complessità e la flessibilità delle regole di accesso, garantite per legge ai pazienti, richiedono che l'apposito portale per pazienti funga da Policy Manager che utilizzi una API nel Policy Repository, per aggiungere, richiedere, aggiornare e eliminare regole d'accesso. Per questo caso di applicazione non sono previsti standard di operabilità

The Swiss Electronic Health Record (EPD) depends on an IHE XDS and multi-community based system where the patient not only consents to the creation and use of the record, but does so by explicitly defining access rules through a patient portal.

The patient's privacy choices (concerning access to his health record) are stored by the community where the patient has established his EPD (reference community) and MUST be respected by all participating systems. It has been specified for the Document Registries to act as Policy Enforcing Service Providers in terms of a XACML PEP. However, as the rules to be enforced MAY not be available to the Document Registry of a community, the XACML PDP needs to be implemented as its own separated actor to establish interoperability regarding policy enforcements. Furthermore, Policy Repositories themselves (XACML PAP) are specified to act as a Policy Enforcing Service Provider.

The complexity and flexibility of access rule definitions that were granted to patients by law, require the Patient Portals to act as Policy Managers that use an API into Policy Repositories to add, query, update and delete policies. There is a lack of interoperability standards regarding this use case.

1.1 Definitions of terms

1.1.1 Electronic Patient Record (EPR)

The object of the Federal Act on Electronic Patient Records (EPRA) is to define the conditions for processing data and documents relating to electronic health records. Using electronic health records, healthcare professionals can access data relevant to treatment of their patients that was compiled and decentrally recorded by healthcare professionals involved in the treatment process. Healthcare professionals may save this data if necessary in their practice and hospital information systems outside of the electronic health records. To access electronic health records, healthcare professionals must join a certified community, which is an association of healthcare professionals and their institutions, and their patients must grant them the necessary access rights. In addition, the electronic health record also allows patients to view their data, to make their own data accessible and to manage the allocation of access rights. Healthcare professionals may only process data in electronic health records with the consent of the patient. Patients have the option of granting individual and graded access rights.
Notation of this term in the following text: **EPR**

1.1.2 EPR circle of trust

From an organizational perspective and in terms of the EPRA, communities are an association of healthcare professionals and their institutions. Communities who want to participate in the Swiss EPR must comply with the certification requirements as laid down in the implementing provisions for the EPRA. Such communities and, in particular, their gateways will be listed in a community portal index provided by the FOPH and therefore form a circle of trust by mutual recognition of their conformity related to data protection and data privacy. Furthermore, all required central services are also part of this circle of trust.
Notation of this term in the following text: **EPR circle of trust**

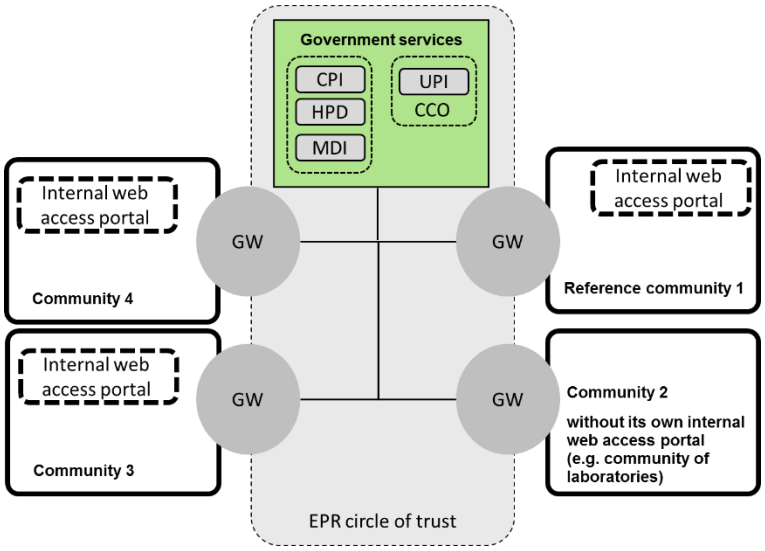

Figure 1: Swiss EPR circle of trust

Legend:

- GW: Gateway
- CPI: Community / Portal Index
- UPI: Unique Person Identification
- HPD: Healthcare Provider Directory
- MDI: Metadata Index-Service

1.1.3 Reference community

If a patient decides to open an EPR, she or he first chooses a community that manages all of his current consents and access right configurations to be used by other EPR users (in essence healthcare professionals) while accessing his personal EPR. Consents and access rights for one patient are managed by exactly one community in the EPR circle of trust.
Although the term home community is used by IHE in a slightly different way, the current specification states this consent and access right management community as reference community.
Cross-community accesses to documents within the EPR are only permitted when the initiating user gets permission by the access rights defined by the patient. Although cross-community accesses may occur between each community within the EPR circle of trust regardless whether it is the patient's reference community or not, the responding community must always apply the current access right settings managed by the reference community.
The patient may change his reference community at any time (for example, when moving to another residence).
Notation of this term in the following text: **referenceCommunity**

1.1.4 Patient Identifiers (EPR-PID, MPI-PID)

Communities in the EPR circle of trust use the national EPR patient identifier (EPR-PID) only for cross-community communication. The federal Central Compensation Office (CCO)[1] is the institution which issues EPR-PID's (EPR Sectorial Personal Identification Number). CCO is the only institution which is allowed to correlate the Social Security Number (AVN13) with the EPR-PID. There is no correlation possible back from the EPR-PID to the Social Security Number. This is political intention in order to achieve highest possible patient privacy.
Within a community patients are identified by a MPI-PID which is managed by a community Master Patient Index (MPI). Primary Systems may correlate their local patient identifier with the MPI-PID. For cross-community communication the gateways may correlate the MPI-ID to the EPR-PID.

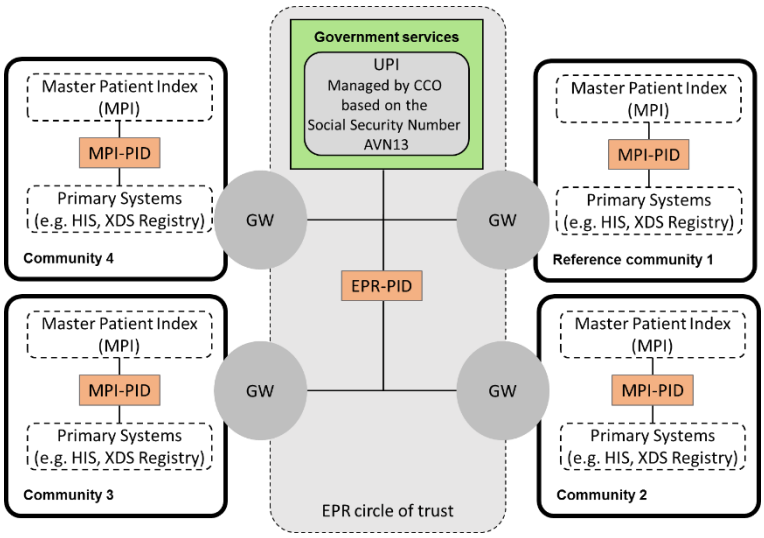

Figure 2 Swiss Patient Identifiers

1.1.5 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

[1] http://www.zas.admin.ch/index.html

2 Volume 1 – Integration Profiles

2.1 Overview

The **Authorization Decision Request (ADR)** may be understood as a subsequent process to IHE XUA. XUA formulates the user's identity (SAML assertion) that is trying to access data through a corresponding transaction. ADR takes the information provided by the identity assertion of a transaction and formulates a decision request query by a description of the subject (who), action (how), resource (what) and environment (when). The response contains an access decision for each resource.
The **Privacy Policy Query (PPQ)**, however, may rather be understood similar to XDS transactions. A Policy Manager applies PPQ transactions to add, query, update and delete policies held by the Policy Repository. PPQ is the pre-requisite for Patient Portals to manipulate the policies, authorization decisions are finally based on. It is important to understand that PPQ transactions underlie the same access control mechanisms as XDS transactions do. Therefore XUA identity assertions MUST be provided, so that the Policy Repository can verify (through a subsequent ADR transaction) whether the access control mechanism allows the changes.

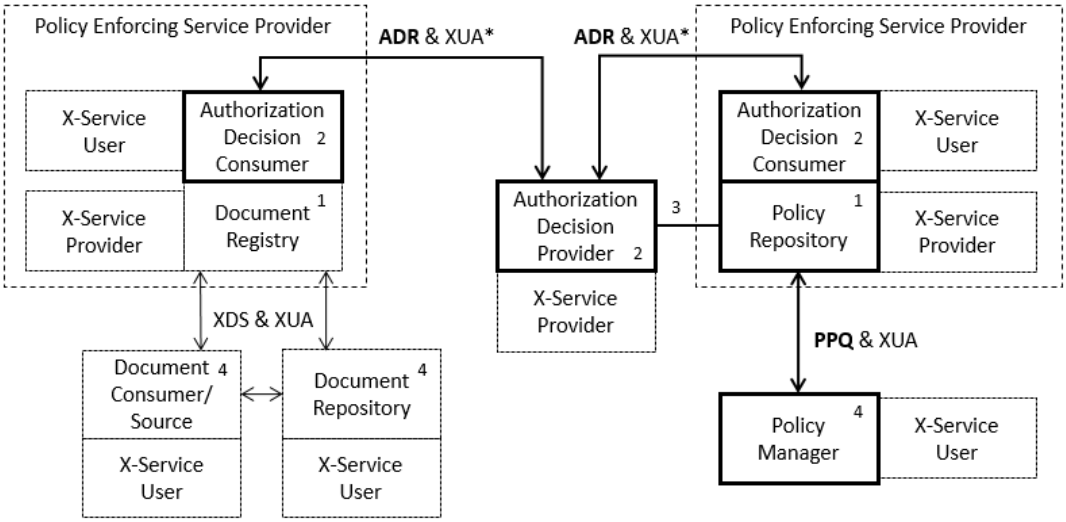

*Figure 3: ADR and PPQ Actors - shows the actors directly involved in the ADR and PPQ Profile and the relevant transactions between them. If needed for context, other actors that MAY be indirectly involved due to their participation in other related profiles are shown in dotted lines. Actors which have a mandatory grouping are shown in conjoined boxes. *) The ADR transaction MUST provide a XUA identity assertion of the current user mainly for auditing reasons.*

1. Document Registries, Repositories and Policy Repositories MUST be grouped with the ADR Authorization Decision Consumer and XUA X-Service Provider actors to become Policy Enforcing Service Providers.
2. ADR transactions are protected by XUA as well, which requires the Authorization Decision Consumer to be grouped with the X-Service User actor and the Authorization Decision Provider to be grouped with the X-Service Provider actor (marked with *)
3. The ADR Authorization Decision Provider SHOULD be grouped with a Policy Repository or requires privileged access to the policies stored by the Policy Repository.
4. A Policy Manager applies PPQ transactions to add, query, update and delete policies stored by the Policy Repository. Document Consumers apply XDS Registry Stored Query transactions to retrieve document metadata. Document Repositories apply XDS Register Document Set transactions due to XDS Provide and Register transactions by a Document Source. All three are grouped with the XUA X-Service User Actor.

2.2 EPR XUA Requirements for XDS and PPQ

A SAML 2.0 **<Assertion>** is added to the WS-Security context of the SOAP Header of each transaction message to communicate entities (user identities) that initiated those transactions. This is a prerequisite for subsequent Authorization Decision Query Requests.

```
<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
  <saml2:Assertion xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:xs="http://www.w3.org/2001/XMLSchema" ID="_37d8092df99f08cd8435ac29a7062092"
    IssueInstant="2014-04-09T19:10:00.294Z" Version="2.0">
     <!--Identity Claims-->
  </saml2:Assertion>
</wsse:Security>
```

Listing 1: The WS Security context of the SOAP header with the SAML2 Assertion element. For simplicity the identity claims are not shown.

The EPR SAML 2.0 <Assertion> has the child elements **<Issuer>**, **<Signature>**, **<Subject>**, **<Conditions>, <AuthnStatement>** and **<AttributeStatement>**. The <AttributeStatement> element caries a number of attributes that reflect the identity claims being made.
The EPR requires the following details to be claimed within the assertion:

<Issuer> the system that issued the token and therefore confirms that the identified user was properly authenticated and that the attributes included in the token are accurate. For further details see [SAML 2.0].
<saml2:Issuer>**urn:e-health-suisse:xua:community:ksa**</saml2:Issuer>

<Signature> an X.509 signature by a trusted entity (XUA Assertion Provider) to guaranty the confidentiality of the claims being made and unaltered content of the assertion. For further details see [SAML 2.0].

```
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
          <ds:Reference URI="#_37d8092df99f08cd8435ac29a7062092">
            <ds:Transforms>
              <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
              <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
             <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="xs"/>
              </ds:Transform>
            </ds:Transforms>
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue>NRrlqwGn8o9tO0DIkYbOaXNqlM0=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
<ds:SignatureValue>dbBafjF2NPY0WztQvRpda5DOV8BrPYL5KlCx8yvnEBZ9TQrKnjwhcE=</ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>
        <!-- X.509 Certificate -->
      </ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
```

Listing 2: The Signature Element of the WS Security context providing the details of signature algorithm used. For simplicity the X.509 certificate is not shown.

<Subject> identifies the Requester Entity (Who is asking for access?). This element SHALL have the following SAML 2.0 **<NameID>** child element with the following attributes:

@Format=" urn:oasis:names:tc:SAML:2.0:nameid-format:persistent" and **@NameQualifier="urn:e-health-suisse:epd-pid"** in case of a patient or **@NameQualifier="urn:gs1:gln"** in case of a professional or **@NameQualifier="urn:e-health-suisse:custodian-id"** in case of a custodian or guardian, who's been assigned to manage a patient's Health Record.
The Value of this element SHALL convey the subject identifier.

<Subject> SHALL have a second child element **<SubjectConfirmation>** with the following attribute:
@Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"

```
<saml2:Subject>
  <saml2:NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent"
                NameQualifier="urn:gs1:gln">7601000000000</saml2:NameID>
  <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"/>
</saml2:Subject>
```

Listing 3: Subject element of the SAML assertion providing the ID and the name qualifier of the requesting subject.

<Conditions> specifying a validity period (time stamps) to prevent "replay" of the assertion while attributes MAY have changed. The time period MUST be defined between a minimum of 5 seconds and a maximum of 10 minutes.
An audience restriction (urn:e-health-suisse:token-audience:all-communities) specifies the intended recipient or system the assertion SHALL be valid for.
The reuse of the token (signed SAML identity assertion) MAY be denied by setting a <OneTimeUse> element. For further details see [SAML 2.0].

```
<saml2:Conditions NotBefore="2016-02-09T19:10:00.294Z" NotOnOrAfter="2016-02-09T19:15:00.294Z">
  <saml2:AudienceRestriction>
    <saml2:Audience>urn:e-health-suisse:token-audience:all-communities</saml2:Audience>
  </saml2:AudienceRestriction>
</saml2:Conditions>
```

Listing 4: The condition element of the SAML 2 assertion defining the assertion life time.

<AuthnStatement> specifying the authentication procedure by which the entity's identity (e.g. a user) was verified. For further details see [SAML 2.0].

```
<saml2:AuthnStatement AuthnInstant="2016-02-09T19:10:00.294Z">
  <saml2:AuthnContext>
<saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTrans-
port</saml2:AuthnContextClassRef>
  </saml2:AuthnContext>
</saml2:AuthnStatement>
```

Listing 5: The authentication statement providing the authentication procedure used by the requesting system.

<AttributeStatement> identifies the Requester Entity's attributes / identity claims. There are six mandatory **<Attribute>** child elements as follows.
There SHALL be one <Attribute> element with the attribute:
@Name="urn:oasis:names:tc:xspa:1.0:subject:subject-id"
The **<AttributeValue>** child element SHALL convey the subject's real world name as plain text as defined by IHE XUA.

There SHALL be one <Attribute> elements with the attribute:
@Name="urn:oasis:names:tc:xacml:2.0:subject:role"
The **<AttributeValue>** child element SHALL convey a coded value of the subject's **<Role>**. There are four roles to be distinguished within the EPR: "Patient(in)", "Behandelnde(r)", "Hilfsperson" and "Stellvertreter(in)".

There SHALL be one or more <Attribute> elements with the attribute:
@Name="urn:oasis:names:tc:xspa:1.0:subject:organization"
The **<AttributeValue>** child element SHALL convey a plain text the subject's organization is named by.

There SHALL be one or more <Attribute> elements with the attribute:
@Name="urn:oasis:names:tc:xspa:1.0:subject:organization-id"
The **<AttributeValue>** child element SHALL convey the ID of the subject's organization or group. That ID MUST be an OID in the format of an URN. The OIDs of organizations or groups are stored within the EPR's Healthcare Provider Directory (aka HPI/HOI).

There SHALL be an <Attribute> element with the attribute:
@Name="urn:oasis:names:tc:xacml:2.0:resource:resource-id"
The **<AttributeValue>** child SHALL convey the EPR-PID identifier of the patient's record the current transaction is related to.
(syntax as used in iti-18 XDSDocumentEntryPatientId)

.

There SHALL be an <Attribute> element with the attribute:
@Name=" urn:oasis:names:tc:xspa:1.0:subject:purposeofuse"
The **<AttributeValue>** child element SHALL convey a coded value of the current transaction's **<PurposeOfUse>**. There are two values to be distinguished within the EPR: "Normalzugriff", "Notfallzugriff" (displayName).

```
<saml2:AttributeStatement>
  <saml2:Attribute Name="urn:oasis:names:tc:xspa:1.0:subject:subject-id">
    <saml2:AttributeValue>Hans Muster</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute Name="urn:oasis:names:tc:xacml:2.0:subject:role">
    <saml2:AttributeValue>
      <Role xmlns="urn:hl7-org:v3" xs:type="CE“
            code=“PAT"
            codeSystem="2.16.756.5.30.1.127.3.10.4"
            codeSystemName="eHealth Suisse EPR Akteure"
            displayName=“Patient(in)"/>
    </saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute Name="urn:oasis:names:tc:xspa:1.0:subject:organization">
    <saml2:AttributeValue>Kantonspital Aarau</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute Name="urn:oasis:names:tc:xspa:1.0:subject:organization-id">
    <saml2:AttributeValue>urn:oid:2.999</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute Name="urn:oasis:names:tc:xacml:2.0:resource:resource-id">
   <saml2:AttributeValue>8901^^^&amp;amp;2.16.756.5.30.1.127.3.10.3&amp;amp;ISO</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute Name="urn:oasis:names:tc:xspa:1.0:subject:purposeofuse">
    <saml2:AttributeValue>
      <PurposeOfUse xmlns="urn:hl7-org:v3" xs:type="CE"
                    code=“NORM"
                    codeSystem="2.16.756.5.30.1.127.3.10.5"
                    codeSystemName="eHealth Suisse Verwendungszweck"
                    displayName=“Normalzugriff"/>
    </saml2:AttributeValue>
  </saml2:Attribute>
</saml2:AttributeStatement>
```

Listing 6: The SAML 2 attribute statement with the IHE XUA attribute claims.

2.3 Authorization Decision Request (CH:ADR)

This supplement defines new functionalities for XDS-based communities concerning the enforcement of access policies. They are applied to the clinical data stored by an XDS Document Registry, as well as to the access policies themselves, which are stored in a Policy Repository.

2.3.1 Motivation

The Document Registry, as the only system with knowledge of all clinical documents (and which only exists once) within communities (affinity domains), is generally thought of as an appropriate actor to enforce access rules on stored metadata. It is common that the Document Registry is inherently combined with the ability to make authorization decisions, which postulates access to the rules to be enforced and the ability to interpret them. As this is not necessarily given in all XDS environments, a separation of actors for decision making and enforcement, as well as the development of corresponding transactions greatly enhances interoperability. This is by no means a new idea, as the XACML standard as well as existing IHE profiles (SeR) envision the same concept and therefore will be adopted and adapted by ADR.
More generally, ADR enables a policy enforcing service provider (e.g. a Document Registry or a Policy Repository) to retrieve access decisions from an authority with access to the rules and the ability to interpret them.

2.3.2 Objectives and Constraints

The objective of the ADR Profile is the definition of a mechanism to request authorization decisions and convey the results between the actors "Authorization Decision Consumer" and "Authorization Decision Provider". Both are to be interpreted as specific implementations of PEP and PDP as defined by the XACML specification. There is a considerable overlap of concepts and use cases with the existing IHE Secure Retrieve (SeR) Profile. The following specification is based on IHE SeR, which was adapted to the needs of the actors and use cases of ADR. Transport, transaction types and content shall be based on the same standards and technologies as far as possible.
Two new actors and a new ADR-specific Authorization Decision Query transaction are being introduced. This profile describes how a Policy Enforcing Service Provider can request authorization decisions on certain resources and actions depending on user entities, a patient's record and other parameters allowed by the underlying standards.
Summarized, the constraints upon which this profile is developed are:

- The XACML data-flow model serves as the underlying processing model.
- There are Authorization Decision Providers acting as XACML PDPs with access to the policies and the capability to perform access decisions on.
- The policies are stored in a Policy Repository acting as XACML PAP.
- Policy enforcing service providers (e.g. Document Registries) act as XACML PEPs by implementing the Authorization Decision Consumer and the corresponding enforcement of a decision.
- The transactions between the profile's actors rely on SAML 2.0 profile of XACML v2.0.
- Policy enforcing service providers are grouped with a XUA X-Service Provider actor and therefore are capable of processing identities communicated in a SAML identity assertion.

2.3.3 Actors / Transactions

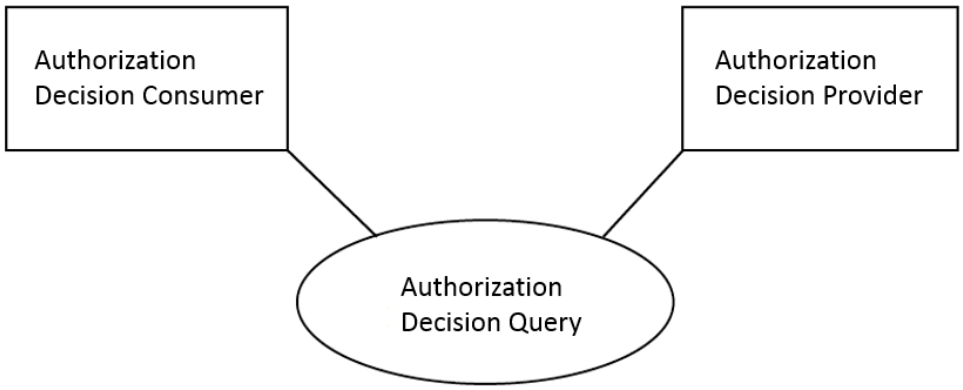

Figure 4: Diagram of actors involved in the ADR profile.

Actor:	Authorization Decision Provider
Role:	This actor accesses and interprets rules/policies and permits or denies access to resources.
Actor:	Authorization Decisions Consumer
Role:	This actor queries for authorization decisions.

Table 1 Actor Roles

2.4 Privacy Policy Query (CH:PPQ)

This supplement defines new functionalities for XDS-based communities concerning the management of access policies in terms of updating or modifying policies as well as querying policies from and adding policies to a Policy Repository through a Policy Manager.

2.4.1 Motivation

The EPR defines the Policy Repository to act as XACML PAP that holds the access rules for the entire record as defined by the patient. Communities offering that service can be chosen by the patient to serve as the holder of that information (referenceCommunity). The community also provides a Patient Portal to allow the corresponding management of that information by the patient.
For the EPR, patients have extensive choices regarding their privacy preferences. There is a base rule stack, which defines a number of general access levels; the patient has a choice to grant to individual providers. A corresponding rule stack on top of the base rule stack MUST be allowed for the patient to be created, retrieved, manipulated and deleted. In addition to that, the patient MAY even define who has access not only to the record's documents but also to the patient's access rule stack including the ability to modify it.
The complexity and flexibility REQUIRED, can hardly be facilitated by existing standards. There are simpler approaches existing (e.g. IHE BPPC) to allow the expression of privacy choices by formulating consent to a set of fixed access policies (Allow publishing? Allow access during normal treatment? Allow break-the-glass?). However, allowing the patient to express specific rules for individual documents, providers and organizations requires a richer user experience and the ability to retrieve, change and delete individual rules. This implies using an API approach instead of a document-centric approach.

2.4.2 Objectives and Constraints

The objective of PPQ is the definition of actors and transactions to convey access policies from a Patient Portal to the referenceCommunity. Two new actors "Policy Manager" and "Policy Repository" are introduced. While the Policy Repository may be interpreted as a specific implementation of a XACML

PAP, no analogy to the Policy Manager actor is defined in XACML. Therefore the Policy Manager is being introduced as an entirely new PPQ actor.
This profile describes how Policy Managers query, add, update and delete policies, allowing a Health Record user to manage access rights according to the freedom of choice that was granted to the patient by Swiss regulations.
Constraints upon which this profile is developed are:

- The development of transactions between the profile's actors relies on SAML 2.0 and XACML SAML extension types, elements and protocols as specified in OASIS SAML 2.0 profile of XACML v2.0.
- The Policy Repository itself acts as a Policy Enforcing Service Provider being grouped with a XUA X-Service Provider actor. Therefore it is capable of processing identities communicated in a SAML identity assertion.
- The Policy Repository responds to PPQ Requests according to the result of ADR (transaction is allowed or not allowed to be performed).
- Respectively, Policy Managers are grouped with a XUA X-Service User to convey the current user's identity.

2.4.3 Actors / Transactions

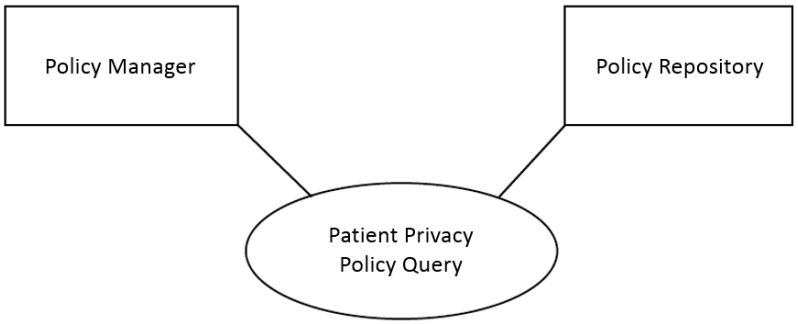

Figure 5: Actors involved in the PPQ profile.

Actor:	Policy Manager
Role:	This actor queries for existing policy sets, adds new policy sets, updates or deletes existing policy sets.
Actor:	Policy Repository
Role:	This actor acts as a XACML Policy Administration Point

Table 2: Actor Roles of the PPQ profile.

3 Volume 2 – Transactions

3.1 Authorization Decision Request (CH:ADR)

3.1.1 Scope

This transaction is used by the Authorization Decisions Consumer to query for authorization decisions, granted and managed by the Authorization Decisions Provider.
The Authorization Decisions Consumer asks for authorizations based on: the requester entity (**Subject**), the **Resources** available to be accessed by the Subject depending on the **Action** that was initiated, each completed by further context parameters.
This transaction is based on SOAP v1.2 exchange protocol and Synchronous Web services.

3.1.2 Referenced Standards

- W3C SOAP Version 1.2
  https://www.w3.org/TR/soap12/
- Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0
  https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

The home page of the "OASIS eXtensible Access Control Markup Language" technical committee: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml references all XACML related protocols and specifications for implementers of this profile. Those are:

- OASIS Multiple Resource Profile of XACML v2.0
  https://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-mult-profile-spec-os.pdf
- OASIS Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of SAML v2.0 for Healthcare Version 2.0 (not normative)
  https://docs.oasis-open.org/xspa/saml-xspa/v2.0/saml-xspa-v2.0.html
- OASIS eXtensible Access Control Markup Language (XACML) v2.0
  (Original: https://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf)
  Please be aware of the errata of the specification document as published on the XACML technical committee home page:
  Errata: http://www.oasis-open.org/committees/download.php/24548/access_control-xacml-2.0-core-spec-os-errata.zip (spec and schema)
- OASIS SAML 2.0 profile of XACML v2.0
  (Original: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf)
  Please be aware of the errata of the specification document as published on the XACML technical committee home page:
  Errata: www.oasis-open.org/committees/download.php/24681/xacml-profile-saml2.0-v2-spec-wd-5-en.pdf

3.1.3 Interaction Diagram

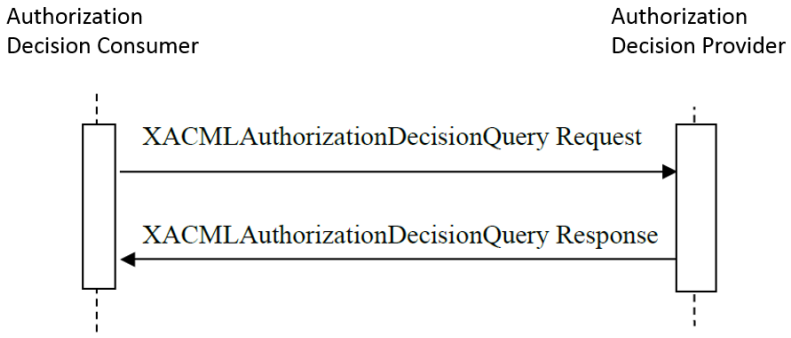

Figure 6: Sequence diagram of the XACMLAuthzDecisionQuery transaction of the ADR profile.

3.1.4 XACMLAuthzDecisionQuery Request

This message enables the Authorization Decisions Consumer to query the Authorization Decisions Provider for authorizations. This message relies on the SAML v2.0 extension for XACML and uses the element <XACMLAuthzDecisionQuery> to convey the Resource metadata, Subject identifier and Actions. The Authorization Decisions Consumer can ask for authorization regarding a number Resources in one query as the request message complies with the Multiple Resource Profile of XACML v2.0. Actors involved support XUA and use SAML identity assertions to identify entities (see ITI TF-1: 39.5 and 39.6). SAML attribute elements SHALL be mapped into XACML context attribute elements as defined in SAML 2.0 profile of XACML v2.0.

3.1.5 Trigger Events

The Authorization Decision Consumer of the EPR sends this message when it needs to verify whether there is an authorization to disclose specific Resources to an entity requesting them; e.g. to allow or deny access to and the manipulation of policies stored by a policy repository or to allow or deny access to document metadata stored in a Document Registry based on the entry's confidentiality code. In addition to that the Authorization Decision Consumer of the EPR sends this message when it needs to verify whether there is an authorization to persist specific Resources e.g. to allow or deny storage of document metadata in a Document Registry based on the entry's confidentiality code. The trigger events are:

- The grouped XDS Document Registry receiving a Registry Stored Query Request [ITI-18] and a Provide X-User Assertion [ITI-40] transaction, that identifies the specific requester entity within a SAML assertion, from an XDS Document Consumer;
- The grouped XDS Document Registry receiving a Register Document Set-b [ITI-42] and a Provide X-User Assertion [ITI-40] transaction, that identifies the specific requester entity within a SAML assertion, from an XDS Document Repository;
- The grouped PPQ Policy Repository receiving a Privacy Policy Query transaction (see this document) and a Provide X-User Assertion [ITI-40] transaction from a PPQ Policy Manager that identifies the specific requester entity within a SAML assertion.

3.1.6 Message Semantics

3.1.6.1 ADR due to XDS Registry Stored Query [ITI-18]

For the XDS Registry Stored Query related access decision enforcement, the EPR relies on the XDS Confidentiality Code within the document metadata to be accessed to represent a subset of the patient's health record. The Authorization Decisions Consumer MUST create one request to query for an access decision for each subset (rather than the actual document metadata objects), before providing the corresponding document metadata to a consumer. Therefore one of the attributes of each Resource within the Request must be a XDS confidentiality code defining the subset for an access decision to be made on (details below).

ADR due to XDS Register Document Set-b [ITI-42]

For the XDS Register Document Set related access decision enforcement, the EPR relies on the XDS Confidentiality Code within the document metadata to be stored in the patient's Health Record. The Authorization Decisions Consumer (Document Registry) MUST create one request to query for an access decision for each Confidentiality Code, before allowing the Register transaction to a Document Repository. One of the attributes of each Resource within the Request must be a XDS confidentiality code for an access decision to be made on (details below).

3.1.6.2 ADR due to PPQ

The EPR allows patients and their guardians to manage the patient's Health Record access rights. In addition to that, the patient may allow a professional to delegate his access rights to another professional if necessary.

In the case of ADR due to PPQ an access decision must be requested for each actual object (Resource) that access is being requested for (not a class of objects as it is the case for ADR due to XDS). Each Resource represents a policy set that's being queried, added, deleted or updated by a PPQ transaction. An access decision is to be requested for each of these Resources before the corresponding action can be granted (or has got to be denied, depending on the decision).

A professional may only delegate access rights to another professional not exceeding her or his own access level that was initially granted by the patient. The access level to be granted is encoded within the value of the referenced-policy-set attribute. Therefore, in case of ADR due to PPQ, one of the attributes of each Resource must be a referenced policy set (details below).

3.1.6.3 Semantics

The XACMLAuthzDecisionQuery Request message SHALL use SOAP v1.2 message encoding. The WS-Addressing Action header SHALL have this value:
urn:e-health-suisse:2015:policy-enforcement:AuthorizationDecisionRequest
The recipient of the Authorization Decision Query SHALL be identified by the WS-Addressing To header (URL of the endpoint).
A SAML 2.0 Identity Assertion SHALL be conveyed within the WS-Security Security header.

```
<soap:Envelope
xmlns:soap=http://www.w3.org/2003/05/soap-envelope xmlns:wsa=http://www.w3.org/2005/08/addressing
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:wsse=http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
xmlns:ds=http://www.w3.org/2000/09/xmldsig#
xmlns:xacml-saml="urn:oasis:xacml:2.0:saml:assertion:schema:os“
xmlns:xacml-samlp="urn:oasis:names:tc:xacml:2.0:profile:saml2.0:v2:schema:protocol“
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion“
xmlns:xacml="urn:oasis:names:tc:xacml:2.0:policy:schema:os“
xmlns:epd="urn:e-health-suisse:2015:policy-administration“
xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os“ xmlns:hl7="urn:hl7-org:v3“>
```

```
<soap:Header>
  <wsa:Action>urn:e-health-suisse:2015:policy-enforcement:AuthorizationDecisionRequest </wsa:Action>
  <wsa:MessageID>urn:uuid:e4bb38c7-e546-4bb1-8d68-2bccf783dfbf</wsa:MessageID>
  <wsa:To>https://e-health-suisse-adr-provider.ch</wsa:To>
  <wsse:Security>
    <saml:Assertion>
      <!--SAML Assertion as described above-->
    </saml:Assertion>
  </wsse:Security>
</soap:Header>
<soap:Body>
 <!—ADR TRANSACTION PAY LOAD-->
</soap:Body>
</soap:Envelope>
```

Listing 7: The SOAP envelope with the security header and the transaction payload of the ADR transactions. For better reading placeholder are used for the SAML assertions and the transaction payload.

The body of the message SHALL use an **<XACMLAuthzDecisionQuery>** element (defined in the SAML 2.0 Profile for XACML v2.0) to convey a <Request> with the Authorization Query parameters (Subject, Resource, Action, Environment). This element SHALL contain the following attribute:
@ReturnContext SHOULD be set to "**false**" because the content of the XACMLAuthzDecisionQuery Request is not needed within the Authorization Result.
@InputContextOnly SHALL be set to "**false**", as the Authorization Decision Provider may have further information and rules, other than the parameters included in the request, to determine a decision. This should not be restricted by the Authorization Decision Consumer.
This profile does not define further constraints for other attributes of this element (see OASIS SAML 2.0 profile of XACML v2.0 for details).

```
<soap:Body>
  <xacml-samlp:XACMLAuthzDecisionQuery InputContextOnly="false" ReturnContext="false"
  ID="_682fee8b-46c0-442a-8c54-fd9d656412fc" Version="2.0" IssueInstant="2016-02-09T09:30:10.5Z">
    <xacml-context:Request>
      <!--Request Parameters-->
    </xacml-context:Request>
  </xacml-samlp:XACMLAuthzDecisionQuery>
</soap:Body>
```

Listing 8: The SOAP body element for the XACMLAuthzDecisionQuery transaction. For better reading a placeholder is used for the request parameter.

The <XACMLAuthzDecisionQuery> element SHALL have only one child element **<Request>.** This element SHALL comply with OASIS Multiple Resource Profile of XACML v2.0. This element SHALL have the XACML child elements **<Subject>, <Resource>, <Action> and <Environment>**. <Request> and all subsequent elements, attributes and values comply to the namespace xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os". The namespace is left out of the following examples for better readability.

```
<soap:Body>
  <XACMLAuthzDecisionQuery>
    <Request>
      <Subject>
        <!--Attributes-->
      </Subject>
      <Resource>
        <!--Attributes-->
      </Resource>
      <Resource>
        <!--There can be more than one Resource-->
      </Resource>…
      <Action>
        <!--Attribute-->
      </Action>
      <Environment/>
    </Request>
  </XACMLAuthzDecisionQuery>
</soap:Body>
```

Listing 9: The schematic payload of the XACMLAuthzDecisionQuery request. For better reading placeholder are used for the XACML request elements.

<Subject> identifies the Requester Entity. It SHALL have at least the following **<Attribute>** child elements:

@AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id" and **@DataType="http://www.w3.org/2001/XMLSchema#string"**.
The **<AttributeValue>** child element SHALL convey the subject identifier. This element SHALL have the same value of the /Subject/NameID element conveyed within the SAML assertion.

@AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id-qualifier" and **@DataType=" http://www.w3.org/2001/XMLSchema#string"**.
The **<AttributeValue>** child element SHALL convey the subject ID qualifier. This element SHALL have the same value as the /Subject/NameID/@NameQualifier conveyed within the SAML assertion, e.g. **urn:e-health-suisse:epd-pid** in case of a patient or guardian or **urn:gs1:gln** in case of a professional or auxiliary person.

@AttributeId="urn:ihe:iti:xca:2010:homeCommunityId" and **@DataType="http://www.w3.org/2001/XMLSchema#anyURI"**.
The **<AttributeValue>** child element SHALL convey the home community id. This value is not necessarily conveyed within the XUA SAML assertion. It SHALL be set to the OID of the Authorization Decision Consumer's community. That ID MUST be an OID in the format of an URN.

@AttributeId="urn:oasis:names:tc:xacml:2.0:subject:role" and **@DataType="urn:hl7-org:v3#CV"**.
The **<AttributeValue>** child element SHALL convey the coded value for the subject's role. This element SHALL have the same value as the /AttributeStatement/Attribute[@name="urn:oasis:names:tc:xacml:2.0:subject:role"]/AttributeValue conveyed within the SAML assertion.

@AttributeId="urn:oasis:names:tc:xacml:2.0:subject:organization-id" and **@DataType=" http://www.w3.org/2001/XMLSchema#anyURI"**.
The **<AttributeValue>** child element SHALL convey the organization identifier. This element SHALL have the same value as the organization-id conveyed within the SAML assertion. urn:gs1:gln

@AttributeId="urn:oasis:names:tc:xacml:2.0:subject:purposeofuse" and **@DataType="urn:hl7-org:v3#CV"**.
The **<AttributeValue>** child element SHALL convey the coded value for the subject's purpose of use. This element SHALL have the same value of the <AttributeStatement>/<Attribute>/<AttributeValue> element @**PurposeOfUse** conveyed within the SAML assertion.

```
<Request>
  <Subject>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
               DataType="http://www.w3.org/2001/XMLSchema#string">
      <Attribute>7601000000000</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id-qualifier"
               DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>urn:gs1:gln</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:ihe:iti:xca:2010:homeCommunityId"
               DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>urn:oid:2.999</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:subject:role“
               DataType="urn:hl7-org:v3#CV">
      <AttributeValue>
        <hl7:CodedValue code=“PAT" codeSystem="2.16.756.5.30.1.127.3.10.4"
        displayName=“PatientIn"/>
      </AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:subject:organization-id"
               DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <Attribute Value>urn:oid:2.999</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:subject:purposeofuse"
               DataType="urn:hl7-org:v3#CV">
      <AttributeValue>
        <hl7:CodedValue code="NORM" codeSystem="2.16.756.5.30.1.127.3.10.5"
        displayName=“Normalzugriff"/>
      </AttributeValue>
    </Attribute>
  </Subject>
  <Resource/>
  <Action/>
  <Environment/>
</Request>
```

Listing 10: Example of the subject attributes elements of the XACMLAuthzDecisionQuery request.

<Resource> identifies the object (ADR due to PPQ) or class of objects (ADR due to XDS) an Authorization Decision is requested for. It SHALL at least have the following **<Attribute>** child elements. The Authorization Decisions Provider MAY ignore any attribute not defined in this specification.

@AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id" and **@DataType="http://www.w3.org/2001/XMLSchema#anyURI"**.
The **<AttributeValue>** child element SHALL convey the resource identifier.

For ADR due to XDS [ITI-18] and [ITI-42] there are always exactly three Resources to be identified, each representing a class of documents: normal, restricted, secret documents. The value MUST be constructed dynamically containing the patient's national identifier extension that was conveyed in the SAML assertion of the XDS transaction identifying the resource (resource-id). The three resource identifiers for ADR due to XDS are:
urn:e-health-suisse:2015:epd-subset:8901:normal,
urn:e-health-suisse:2015:epd-subset:8901:restricted and
urn:e-health-suisse:2015:epd-subset:8901:secret with 8901 as an example value of the patient ID.

For ADR due to PPQ an Authorization Decision MUST be requested for each object itself, not a class of objects. In that case the value is the uuid of a Policy Set the Entity (Subject) is asking access for by a PPQ query, add, update or delete policy, e.g.:
c969c7cd-9fe9-4fdc-83c5-a7b5118922a3.
Therefore, for ADR due to PPQ, there is not a fixed number of <Resource>s (with corresponding Resource IDs) to be specified within the request.

@AttributeId="urn:e-health-suisse:2015:epd-pid" and
@DataType="urn:hl7-org:v3#II".
The **<AttributeValue>** child element SHALL convey the patient's national identifier that was conveyed in the SAML assertion of the XDS transaction identifying the resource (resource-id).

For ADR due to XDS each Resource element MUST also contain the actual confidentiality code corresponding to the resource-id as another attribute:
@AttributeId="urn:ihe:iti:xds-b:2007:confidentiality-code" and
DataType="urn:hl7-org:v3#CV".
The **<AttributeValue>** child element SHALL convey a confidentiality code, e.g.
<hl7:CodedValue code="1051000195109" codeSystem="2.16.840.1.113883.6.96" displayName="normal"/>.

Example for one of the three Resource elements in case of ADR due to XDS [ITI-18]/[ITI-42]:

```
<Request>
  <Subject/>
  <Resource>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
               DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>urn:e-health-suisse:2015:epd-subset:8901:normal</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:e-health-suisse:2015:epd-pid"
               DataType="urn:hl7-org:v3#II">
      <AttributeValue><hl7:InstanceIdentifier root="2.16.756.5.30.1.127.3.10.3"
                      extension="8901"/></AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:ihe:iti:xds-b:2007:confidentiality-code"
               DataType="urn:hl7-org:v3#CV">
      <AttributeValue>
        <hl7:CodedValue code="1051000195109" codeSystem="2.16.840.1.113883.6.96"
        displayName="normal"/>
      </AttributeValue>
    </Attribute>
  </Resource>
  <Resource>
    <!-- resource element for restricted documents corresponding to the example above -->
  </Resource>
  <Resource>
    <!-- resource elements for secret documents corresponding to the example above -->
  </Resource>
  <Action/>
  <Environment/>
</Request>
```

Listing 11: Example of the resource attributes of the XACMLAuthzDecisionQuery request payload. For better reading the part for one confidentiality code is shown in detail, while for the other confidentiality codes placeholders are used.

For ADR due to PPQ each Resource element MUST contain the **referenced** PolicySetId **within** the policy set to be potentially returned, added, updated or deleted (instead of a confidentiality code as in ADR due to XDS).
@AttributeId="urn:e-health-suisse:2015:policy-attributes:referenced-policy-set" and **DataType="urn:hl7-org:v3#CV"**.
The **<AttributeValue>** child element SHALL convey the Policy Identifier that is being referenced within the Policy Set to be queried, added, updated or deleted, e.g. **urn:e-health-suisse:2015:policies:exclusion-list**.

The following example is to clarify this requirement:
If a user (e.g. a patient) tries to add a policy set with ID c969c7cd-9fe9-4fdc-83c5-a7b5118922a3 (as in @AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id") that adds a professional to the exclusion list, the policy set will contain a reference to another policy set from the base configuration, which will have the policy set ID urn:e-health-suisse:2015:policies:exclusion-list. That's the value to be included within the Resource attribute @AttributeId="urn:e-health-suisse:2015: policy-attributes:referenced-policy-set".

Correspondingly, the Resource element of an ADR due to PPQ transaction (to verify if the PPQ user 8901 may be allowed to perform this transaction) SHALL be constructed as in the following example:

```
<Request>
  <Subject/>
  <Resource>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
               DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>c969c7cd-9fe9-4fdc-83c5-a7b5118922a3</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:e-health-suisse:2015:epd-pid"
               DataType="urn:hl7-org:v3#II">
      <AttributeValue><hl7:InstanceIdentifier root="2.16.756.5.30.1.127.3.10.3"
                      extension="8901"/></AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:e-health-suisse:2015:policy-attributes:referenced-policy-set"
               DataType="urn:hl7-org:v3#CV">
      <AttributeValue>urn:e-health-suisse:2015:policies:exclusion-list</AttributeValue>
    </Attribute>
  </Resource>
  <Resource>
    <!—further resource elements-->
  </Resource>
  <Action/>
  <Environment/>
</Request>
```

Listing 12: Example of resource attributes of the XACMLAuthzDecisionQuery request payload for ADR due to PPQ to request an authorization decision for access to the patient's policy configuration.

<Action> identifies the transaction being performed by the Requester Entity. The <Action> element SHALL have one **<Attribute>** child element:

@AttributeId="urn:oasis:names:tc:xacml:1.0:action-id" and
@DataType="http://www.w3.org/2001/XMLSchema#anyURI".
The **<AttributeValue>** child element SHALL convey the action identifier:
urn:e-health-suisse:2015:policy-administration:PolicyQuery or
urn:e-health-suisse:2015:policy-administration:AddPolicy or
urn:e-health-suisse:2015:policy-administration:UpdatePolicy or
urn:e-health-suisse:2015:policy-administration:DeletePolicy for ADR due to PPQ
or
urn:e-health-suisse:2015:action:RegistryStoredQuery for ADR due to XDS ITI-18
or
urn:e-health-suisse:2015:action:RegisterDocumentSet for ADR due to XDS ITI-42.

```
<Request>
  <Subject/>
  <Resource/>
  <Action>
    <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action-id"
               DataType="http://www.w3.org/2001/XMLSchema#anyURI">
      <AttributeValue>urn:e-health-suisse:2015:policy-administration:AddPolicy</AttributeValue>
    </Attribute>
  </Action>
  <Environment/>
</Request>
```

Listing 13: Example of the action setting of XACMLAuthzDecisionQuery request for ADR due to PPQ.

<Environment> The EPD does not specify any <Environment> parameters within the XACMLAuthzDecisionQuery. Therefore this child element MAY be empty: **<Environment />**. The Authorization Decision Provider MAY ignore any attribute in this section when arriving at an authorization decision. However, there is a constraint to the use of <Environment> in case of inputContextOnly of <XACMLAuthzDecisionQuery> was set to true. In that case, current time and date MUST be provided as attributes of <Environment>.

3.1.7 Expected Actions

The Authorization Decisions Provider SHALL return Authorization Decisions that match the XACML Query parameters according to the rules defined in XACML policies.

The Authorization Decision Provider SHALL produce a XACMLAuthzDecisionQuery Response message that conveys the results of the evaluation of the patient's policies against the request. One result for each Resource SHALL be included in the response message.

3.1.8 XACMLAuthzDecision Response

The XACMLAuthzDecision Response message is created by the Authorization Decisions Provider in response to the XACMLAuthzDecisionQuery Request. This message conveys to the Authorization Decisions Consumer the results of the evaluation made by the Authorization Decisions Provider. For each Resource specified within the Request message, the Authorization Decisions Provider provides an Authorization Result that SHALL be used by the Authorization Decisions Consumer to determine which of the requested objects are to be returned or transactions to be allowed in response to the corresponding initial transactions. This message relies on OASIS SAML 2.0 profile of XACML v2.0 protocol standard. Authorization Results are conveyed using the XACMLAuthzDecisionStatement.

3.1.9 Trigger Events

This message is created by the Authorization Decisions Provider after the evaluation of the XACMLAuthzDecisionQuery Request message. The Authorization Decision Provider MUST only return Authorization Decisions applicable to the request.

3.1.10 Message Semantics

The XACMLAuthzDecision Response message is based on OASIS SAML 2.0 profile of XACML v2.0. The WS-Addressing Action header of the SOAP message SHALL be:
urn:e-health-suisse:2015:policy-enforcement:XACMLAuthzDecisionResponse

The **XACMLAuthzDecision <Assertion>** as specified in OASIS SAML 2.0 Profile of XACML v2.0 (Chapter 4.10), is conveyed within a n **XACMLAuthzDecision <Response>**. The Assertion does not need to be signed.
The **<Issuer>** of the Authorization Assertion MUST identify the Authorization Decisions Provider. For the EPR this ID is specified to be the home community ID of the Authorization Decision Provider community encoded as an URN, e.g.
<saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.999.1.1</saml:Issuer>.
In case of all Resources resulting in a decision of "Indeterminate" (details below), the SAML /Status/**StatusCode** of the Assertion shall be the same as the /Result/Status/StatusCode/@Value of the Response: urn:e-health-suisse:2015:error:not-holder-of-patient-policies. Otherwise the SAML /Status/**StatusCode** of the Assertion SHALL be supplied as defined in section " 4.10 Element <samlp:Response>: XACMLAuthzDecision Response" of OASIS SAML 2.0 profile of XACML v2.0.

```
<soap:Body>
 <samlp:Response ID="4v7a68d0-5d67-557e-def4-8e5858676abc3" Version="2.0"
                 IssueInstant=" 2016-02-09T09:30:10.5Z ">
  <saml:Assertion ID="3b5a66d0-5d86-477e-afc4-8e561084edc1" Version="2.0"
                 IssueInstant=" 2016-02-09T09:30:10.5Z ">
   <saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.999.1.1</saml:Issuer>
   <saml:Status>
     <samlp:StatusCode>urn:oasis:names:tc:SAML:2.0:status:Success</samlp:StatusCode>
   </saml:Status>
   <saml:Statement xsi:type="xacml-saml:XACMLAuthzDecisionStatementType">
     <xacml-context:Response>
       <!--Decision Result per Resource-->
     </xacml-context:Response>
   </saml:Statement>
  </saml:Assertion>
 </samlp:Response>
</soap:Body>
```

Listing 14: Schematic payload of the XACMLAuthzDecision response. For better reading the details of the response is suppressed and shown in the listings below.

As specified in the OASIS multiple resource profile of XACML v2.0, the XACML **<Response>** element SHALL contain a **<Result>** element for each **<Resource>** element contained within the XACMLAuthzDecisionQuery Request message. Each <Result> element SHALL contain a @**ResourceId** attribute that identifies which Resource an Access Decision belongs to. A child element **<Decision>** holds the actual decision value.
In case of the decision code of a Result equaling to "Deny", "Permit" or "NotApplicable", the /Result/Status/**StatusCode**/@Value attribute SHALL equal to "urn:oasis:names:tc:xacml:1.0:status:ok". In case of "Indeterminate" it SHALL equal to "urn:e-health-suisse:2015:error:not-holder-of-patient-policies".
<Response> and all subsequent elements, attributes and values comply to the namespace xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os". The namespace is left out of the following examples for better reading purposes.

```
<Response>
  <Result ResourceId="e693657c-50be-46a6-bdcd-05269147f357">
    <Decision>Deny</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
  <Result ResourceId="1c9fa73c-2b9c-41b2-a814-f9164e073c15">
    <Decision>Permit</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
  <Result ResourceId="c969c7cd-9fe9-4fdc-83c5-a7b5118922a3">
    <Decision>Permit</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
</Response>
```

Listing 15: Structure for a response to an ADR due to PPQ request.

```
<Response>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:normal">
    <Decision>Permit</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:restricted">
    <Decision>Permit</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:secret">
    <Decision>NotApplicable</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
    </Status>
  </Result>
</Response>
```

Listing 16: Structure of a Response to an ADR due to XDS message if 8901 was the patient ID (EPR-PID) of the Health Record to be accessed.

As defined in the XACML v2.0 standard, there are four possible values associated with the **<Decision>**. The Authorization Decisions Provider shall use these values as described below:

- **Permit**: if the evaluation was successful and the Subject is authorized to perform the Action on the Resource;
- **Deny:** if the evaluation was successful and the Subject is explicitly not authorized to perform the Action on the Resource.
- **NotApplicable**: if the evaluation was successful, but the Subject is not authorized to perform the Action on the Resource. E.g. a Permit decision can be determined on the Resource "normal access", but no permit or deny decision can be determined for the other resources in the request. The decision code for the other resources MUST be NotApplicable.
- **Indeterminate**: if the evaluation succeeded, but access to the requested Resource is not managed by the Authorization Decisions Manager, or if the evaluation failed. The EPR specifically defines this decision code to be returned, if access rights for a given patient are not managed in the associated Policy Repository and therefore cannot be determined by the Authorization Decision Provider. To distinguish between those two cases, clients may evaluate the /Result/Status/StatusCode/@Value attribute, which has to equal "urn:e-health-suisse:2015:error:not-holder-of-patient-policies" if the Policy Repository is not responsible for holding the given patient policies.

```
<Response>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:normal">
    <Decision>Indeterminate</Decision>
    <Status>
      <StatusCode Value="urn:e-health-suisse:2015:error:not-holder-of-patient-policies"/>
      <StatusMessage>Gemeinschaft ist nicht die Stammgemeinschaft des Patienten</StatusMessage>
    </Status>
  </Result>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:restricted">
    <Decision>Indeterminate</Decision>
    <Status>
      <StatusCode Value="urn:e-health-suisse:2015:error:not-holder-of-patient-policies"/>
      <StatusMessage>Gemeinschaft ist nicht die Stammgemeinschaft des Patienten</StatusMessage>
    </Status>
  </Result>
  <Result ResourceId="urn:e-health-suisse:2015:epd-subset:8901:secret">
    <Decision>Indeterminate</Decision>
    <Status>
      <StatusCode Value="urn:e-health-suisse:2015:error:not-holder-of-patient-policies"/>
      <StatusMessage>Gemeinschaft ist nicht die Stammgemeinschaft des Patienten</StatusMessage>
    </Status>
  </Result>
</Response>
```

Listing 17: The response to a XACMLAuthzDecisionQuery in the case when the patient's policies are not known in the requested community, i.e. when the requested community is not the patients referenceCommunity.

3.1.11 Expected Actions

When the Policy Enforcing Service Provider receives a XACMLAuthzDecisionQuery Response, it SHALL enforce the decision results according to the following EPR policy.

If a **Deny** or **NotApplicable** decision is returned, the

- XDS Document Registry SHALL not disclose the related document metadata in response to ITI-18;
- XDS Document Registry SHALL not store any document metadata from a submission set containing a document that has a confidentiality code for which such a decision was returned and return a XDS registration failure to the XDS Document Repository in response to ITI-42;
- PPQ Policy Repository SHALL not allow the initial PPQ transaction, respectively not return the policy data or make the requested changes to the policies. For add, update and delete transaction there is no partial success defined. If the PPQ transaction includes more than one policy to be added (updated or deleted) and one of the resources is not permitted to be consumed, the entire PPQ request MUST not be allowed.

If a **Permit** decision is returned, the

- XDS Document Registry SHALL disclose the document metadata with the given confidentiality code in response to ITI-18;
- XDS Document Registry SHALL perform the initiated transaction for a submission set containing documents with a corresponding confidentiality code as long as all of the documents of a submission set have a confidentiality code that was permitted by the ADR Response (otherwise see "Deny or NotApplicable" above);
- PPQ Policy Repository shall perform the initiated transactions, respectively return the policy data that has been queried for. For add, update and delete, the decision for all resources within the request MUST be a permit for the PPQ request to be allowed as there is no partial success defined.

If **Indeterminate** is returned, the

- XDS Document Registry MUST request a decision from another Authorization Decisions Provider (XADR as defined below). If there is no Authorization Decisions Provider that returns Deny, NotApplicable or Permit, then the Document Registry SHALL not disclose any document metadata in response to ITI-18 or not perform the ITI-42 transaction respectively.
- PPQ Policy Repository SHALL not allow the initial PPQ transaction, respectively not return the policy data or make the requested changes to the policies.

3.1.12 Enforcement of XDS Retrieve Document Set transactions

The Retrieve of a document MUST be enforced according to the access rights formulated by the patient. If the document metadata of a document cannot be accessed by a user, a Retrieve of the corresponding document MUST be denied by the Document Repository. To implement this functionality, it is recommended for the Document Repositories to initialize a XDS Registry Stored Query [ITI-18] GetDocuments ObjectRef), combined with the XUA Identity Token provided by the Document Consumer [ITI-40], before supplying the document to the Consumer. If the corresponding Document Id is included in the XDS Registry Stored Query Response, the Document SHALL be supplied to the Document Source. If the corresponding Document Id is not included in the XDS Registry Query Response, the Document SHALL NOT be supplied to the Document Source.
The IHE SeR Profile may provide further guidance on the enforcement of access rights concerning the XDS Retrieve Document Set transaction.

3.1.13 Security Considerations

The Authorization Decisions Query transaction requires TLS communication between actors involved. This transaction mandates the creation of Authorizations associated at least with the Requester Entity and with the document metadata (confidentiality code) requested. If additional parameters need to be associated to the authorization, then the same parameters SHALL be provided within the Authorization Decisions Query transaction.

3.1.14 Authorization Decisions Consumer Audit Message

	Field Name	**Opt**	**Value Constraints**
Event	EventID	M	EV (110112, DCM,"Query")
	EventActionCode	M	E = Execute
	EventDateTime	*M*	*not specialized*
	EventOutcomeIndicator	*M*	*not specialized*
	EventTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
Source (Authorization Decisions Consumer) (1)			
Destination (Authorization Decisions Consumer) (1)			
Query Parameters (1..n)			
Requester Entity (1)			
Authorization Result (1..n)			

Source: AuditMessage/ ActiveParticipant	*UserID*	*U*	*not specialized*
	AlternativeUserID	MC	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110153, DCM, "Source")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address, as specified in DICOM PS 3.15 A.5.3.

Destination: AuditMessage/ ActiveParticipant (1)	UserID	M	Authorization Decisions Provider SOAP URI
	AlternativeUserID	U	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110152, DCM, "Destination")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address, as specified in DICOM PS 3.15 A.5.3.

Requester Entity: AuditMessage/ ParticipantObjectIdentification (1)	ParticipantObjectTypeCode	M	"1" (person)
	ParticipantObjectTypeCodeRole	M	"11" (security user entity)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	The Requester Entity (identified in the Attribute with AttributeId

			urn:oasis:names:tc:xacml:1.0:subject:subject-id)
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	*U*	*not specialized*

Query Parameters: AuditMessage/ ParticipantObjectIdentification (1..n)	ParticipantObjectTypeCode	M	"2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"24" (query)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	*M*	*not specialized*
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	M	Resource-ID
	ParticipantObjectDetail	*U*	*not specialized*

Authorization Result: AuditMessage/ ParticipantObjectIdentification (1..n)	ParticipantObjectTypeCode	M	**"**2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"13" (security resource)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	Resource-ID
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	M	Decision Code (Permit, Deny, NotApplicable, Indeterminate)

3.1.15 Authorization Decisions Provider Audit Message

	Field Name	**Opt**	**Value Constraints**
Event	EventID	M	EV (110112, DCM,"Query")
	EventActionCode	M	E = Execute
	EventDateTime	*M*	*not specialized*
	EventOutcomeIndicator	*M*	*not specialized*
	EventTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
Source (Authorization Decisions Provider) (1)			
Destination (Authorization Decisions Provider) (1)			
Query Parameters (1..n)			
Requester Entity (1)			
Authorization Result (1..n)			

Source: AuditMessage/ ActiveParticipant	*UserID*	*U*	*not specialized*
	AlternativeUserID	MC	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110153, DCM, "Source")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address, as specified in DICOM PS 3.15 A.5.3.

Destination: AuditMessage/ ActiveParticipant (1)	UserID	M	Authorization Decisions Provider SOAP URI
	AlternativeUserID	U	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110152, DCM, "Destination")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address, as specified in DICOM PS 3.15 A.5.3.

Requester Entity: AuditMessage/ ParticipantObjectIdentification (1)	ParticipantObjectTypeCode	M	"1" (person)
	ParticipantObjectTypeCodeRole	M	"11" (security user entity)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	The Requester Entity (identified in the Attribute with AttributeId urn:oasis:names:tc:xacml:1.0:subject:subject-id)
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	*U*	*not specialized*

Query Parameters: AuditMessage/ ParticipantObjectIdentification (1..n)	ParticipantObjectTypeCode	M	"2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"24" (query)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	*M*	*not specialized*
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	M	Resource-ID
	ParticipantObjectDetail	*U*	*not specialized*

Authorization Result: AuditMessage/ ParticipantObjectIdentification (1..n)	ParticipantObjectTypeCode	M	**"**2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"13" (security resource)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("ADR", "e-health-suisse", "Authorization Decisions Query")
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	Resource-ID
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	M	Decision Code (Permit, Deny, NotApplicable, Indeterminate)

3.2 Cross-Community Authorization Decision Request (CH:XADR)

Within the EPR, the patient's Health Record access rights are to be stored within the patient's referenceCommunity only. However, each XDS Document Registry MUST act as Policy Enforcing Service Provider, even if the patient's Health Record access rights are not stored within the same community. That means, any Authorization Decision Consumer grouped with a XDS Document Registry SHALL ask each Authorization Decision Provider, even outside their home community, until a response includes a decision code other than NotApplicable. The XADR request follows the same specification as ADR above. Only the service endpoint of an XADR Authorization Decision Provider will be outside of the community of the Authorization Decision Consumer. There may be strategies to be implemented to reduce the number of necessary service calls, which are out of scope of this specification.
For the Authorization Decision Consumer, grouped with a PPC Policy Repository, this is not a requirement, as patient access rights are always managed by a community specific Policy Manager. In that case, the Authorization Decision Provider is always grouped with the Policy Repository of the Policy Managers community, and therefore is the only source of an ADR due to PPC access decision.

3.3 Privacy Policy Query (CH:PPQ)

3.3.1 Scope

These transactions are used by the Policy Manager to add, query, update or delete authorization policies (respectively XACML policy sets) stored in a Policy Repository.
This transaction is based on SOAP v1.2 exchange protocol and Synchronous Web services.

3.3.2 Referenced Standards

- W3C SOAP Version 1.2
  https://www.w3.org/TR/soap12/
- Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0
  https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

The home page of the "OASIS eXtensible Access Control Markup Language" technical committee: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml references all XACML related protocols and specifications for implementers of this profile. Those are:

- OASIS Multiple Resource Profile of XACML v2.0
  https://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-mult-profile-spec-os.pdf
- OASIS eXtensible Access Control Markup Language (XACML) v2.0
  (Original: https://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf)
  Please be aware of the errata of the specification document as published on the XACML technical committee home page:
  Errata: http://www.oasis-open.org/committees/download.php/26986/access_control-xacml-2.0-core-spec-os-errata.doc
- OASIS SAML 2.0 profile of XACML v2.0
  (Original: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf)
  Please be aware of the errata of the specification document as published on the XACML technical committee home page:
  Errata: www.oasis-open.org/committees/download.php/24681/xacml-profile-saml2.0-v2-spec-wd-5-en.pdf

3.3.3 Interaction Diagrams

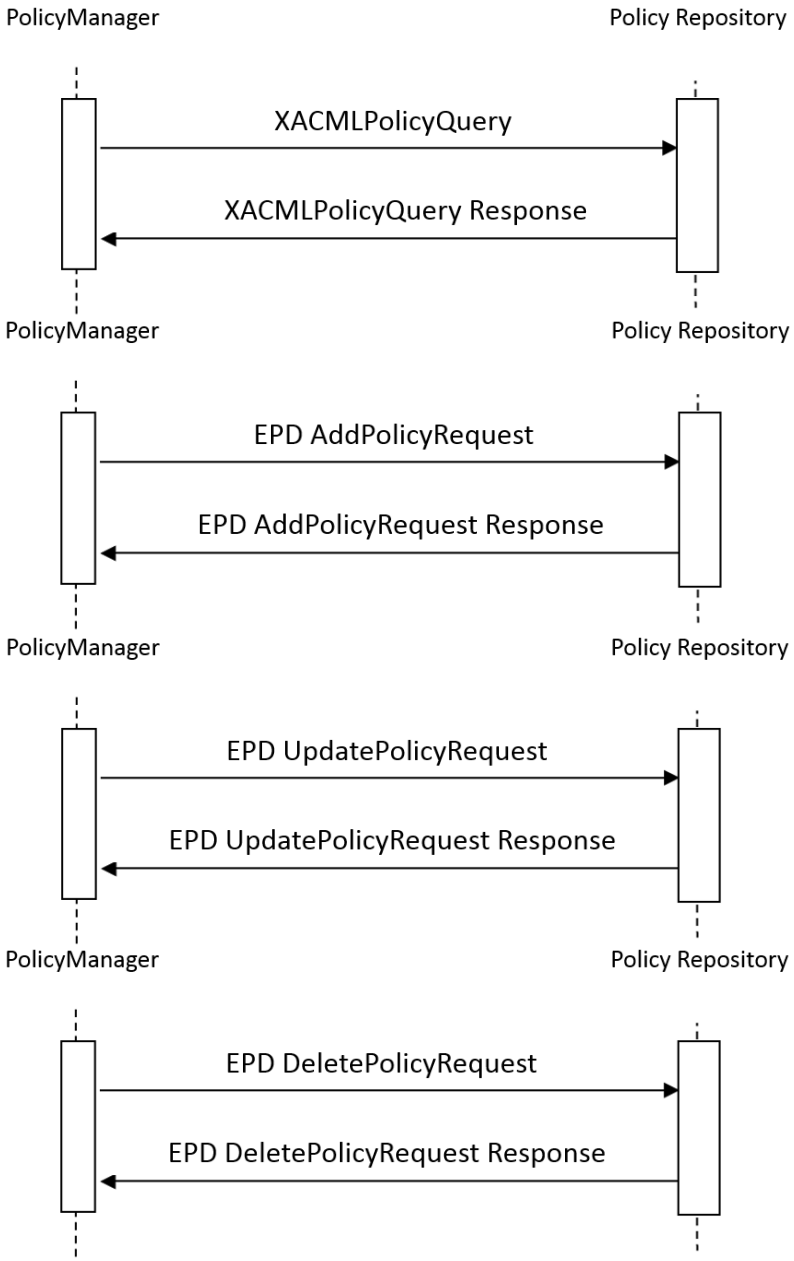

Figure 7: Sequence diagrams for the transactions of the PPQ profile to query, add, update and remove elements of the patient’s privacy policy.

3.3.4 Message Semantics SOAP

PPQ Request messages SHALL use SOAP v1.2 message encoding.
The Addressing Action header of the SOAP message SHALL be:
urn:e-health-suisse:2015:policy-administration:PolicyQuery or
urn:e-health-suisse:2015:policy-administration:AddPolicy or
urn:e-health-suisse:2015:policy-administration:UpdatePolicy or
urn:e-health-suisse:2015:policy-administration:DeletePolicy, depending on the corresponding trigger event.
The recipient of the PPQ Request SHALL be identified by the WS-Addressing To header (URL of the endpoint).
A SAML 2.0 Identity Assertion SHALL be conveyed within the WS-Security Security header.

```
<soap:Envelope
xmlns:soap=http://www.w3.org/2003/05/soap-envelope xmlns:wsa=http://www.w3.org/2005/08/addressing
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:wsse=http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
xmlns:ds=http://www.w3.org/2000/09/xmldsig#
xmlns:xacml-saml="urn:oasis:xacml:2.0:saml:assertion:schema:os“
xmlns:xacml-samlp="urn:oasis:names:tc:xacml:2.0:profile:saml2.0:v2:schema:protocol “
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion“
xmlns:xacml="urn:oasis:names:tc:xacml:2.0:policy:schema:os“
xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os“ xmlns:hl7="urn:hl7-org:v3“
xmlns:epd="urn:e-health-suisse:2015:policy-administration“
xsi:schemaLocation=“urn:e-health-suisse:2015:policy-administration
epd-policy-administration-combined-schema-1.0-local.xsd ws-addr.xsd ">

<soap:Header>
  <wsa:Action>urn:e-health-suisse:2015:policy-administration:PolicyQuery</wsa:Action><!--or-->
  <wsa:Action>urn:e-health-suisse:2015:policy-administration:AddPolicy</wsa:Action><!--or-->
  <wsa:Action>urn:e-health-suisse:2015:policy-administration:UpdatePolicy</wsa:Action><!--or-->
  <wsa:Action>urn:e-health-suisse:2015:policy-administration:DeletePolicy</wsa:Action>
  <wsa:MessageID>urn:uuid:feafcab1-1f9d-4d46-8321-8af925f55f13</wsa:MessageID>
  <wsa:To>https://policy-repository-community-abc.ch</wsa:To>
  <wsse:Security>
    <saml:Assertion>
      <!--XUA SAML Assertion as described above-->
    </saml:Assertion>
  </wsse:Security>
</soap:Header>

<soap:Body>
 <!—PRQ TRANSACTION PAY LOAD-->
</soap:Body>

</soap:Envelope>
```

Listing 18: The SOAP envelope with the security header, the SAML assertions and the transaction payload of the PPQ request. For better reading placeholder are used for the SAML assertions and the transaction payload.

PPQ Response messages SHALL use SOAP v1.2 message encoding.
The Addressing Action header of the SOAP message SHALL be:
urn:e-health-suisse:2015:policy-administration:PolicyQueryResponse or
urn:e-health-suisse:2015:policy-administration:AddPolicyResponse or
urn:e-health-suisse:2015:policy-administration:UpdatePolicyResponse or
urn:e-health-suisse:2015:policy-administration:DeletePolicyResponse, depending on the corresponding trigger event.
The recipient of the PPQ Response SHALL be identified by the WS-Addressing To header (URL of the endpoint).

```
<soap:Envelope
xmlns:soap=http://www.w3.org/2003/05/soap-envelope xmlns:wsa=http://www.w3.org/2005/08/addressing
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:wsse=http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
xmlns:ds=http://www.w3.org/2000/09/xmldsig#
xmlns:xacml-saml="urn:oasis:xacml:2.0:saml:assertion:schema:os“
xmlns:xacml-samlp="urn:oasis:names:tc:xacml:2.0:profile:saml2.0:v2:schema:protocol“
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion“
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:xacml="urn:oasis:names:tc:xacml:2.0:policy:schema:os“
xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os“ xmlns:hl7="urn:hl7-org:v3“
xmlns:epd="urn:e-health-suisse:2015:policy-administration“
xsi:schemaLocation=“urn:e-health-suisse:2015:policy-administration
epd-policy-administration-combined-schema-1.0-local.xsd ws-addr.xsd ">

<soap:Header>
 <wsa:Action>urn:e-health-suisse:2015:policy-administration:PolicyQueryResponse</wsa:Action><!--or-->
 <wsa:Action>urn:e-health-suisse:2015:policy-administration:AddPolicyResponse</wsa:Action><!--or-->
 <wsa:Action>urn:e-health-suisse:2015:policy-administration:UpdatePolicyResponse</wsa:Action><!--or-->
 <wsa:Action>urn:e-health-suisse:2015:policy-administration:DeletePolicyResponse</wsa:Action>
 <wsa:MessageID>urn:uuid:03010066-ba69-43d9-82b1-bb740f8c9a79</wsa:MessageID>
 <wsa:To>https://policy-manager-community-abc.ch</wsa:To>
</soap:Header>

<soap:Body>
 <!—PRQ RESPONSE PAY LOAD-->
</soap:Body>

</soap:Envelope>
```

Listing 19: The SOAP envelope with the transaction payload of the PPQ response. For better reading a placeholder is used the response payload.

3.3.5 XACMLPolicyQuery

This message enables the Policy Manager to query the Policy Repository for existing policies of a patient.
This message relies on SAML 2.0 profile of XACML v2.0 (as specified in 3.3.2 Referenced Standards). Actors involved support XUA and use SAML identity assertions to identify current user entities for subsequent access enforcement.

3.3.6 Trigger Events

The Policy Manager sends this message when it needs to retrieve existing XACML policies or policy sets of a patient stored by a Policy Repository (of the patient's referenceCommunity).

3.3.7 Message Semantics

This message relies on a SAML v2.0 extension protocol element <xacml-samlp:XACMLPolicyQuery> with xmlns:xacml-samlp="urn:oasis:names:tc:xacml:2.0:profile:saml2.0:v2:schema:protocol" (please refer to the referenced specification in 3.3.2.

According to the schema, there are three "ways" to query for Policies or Policy Sets. The one most appropriate for the EPR's Policy Managers is to query for all Policiy Sets of a patient by a xacml-context:Request as defined by xacml-samlp:XACMLPolicyQuery. A Policy Manager MAY also query for policies or policy sets by ID, which SHALL be answered by the Policy Repository. A corresponding example is provided below.

For PPQ, the patient identified by a patient ID is included as a Resource Attribute to be queried for (all policies matching that Resource SHALL be returned if allowed). All XACML <xacml:Policy> and <xacml:PolicySet> instances potentially applicable to this Request MUST be returned.
The <Attribute> element MUST have an AttributeId of urn:e-health-suisse:2015:epd-pid and DataType of urn:hl7-org:v3#II declared. The <AttributeValue> SHALL be an InstanceIdentifier as specified by HL7, identifying the patient's record a PolicySet was formulated to control access for. The Policy Manager MAY query for single Policies too. In that case a Policy ID is required as the <Resource> of the <Request>. A Request MAY contain more than one Resource but there SHALL be "one request per patient", meaning the InstanceIdentifier for a patient's record must occure with one and the same value throughout a XACMLPolicyQuery. <Subject>, <Action> and <Environment> have no PPQ use-case yet.

```
<soap:Body>
  <xacml-samlp:XACMLPolicyQuery>
    <xacml-context:Request>
      <xacml-contect:Subject />
      <xacml-context:Resource>
        <xacml-context:Attribute    AttributeId="urn:e-health-suisse:2015:epd-pid"
                                    DataType="urn:hl7-org:v3#II" >
          <xacml-context:AttributeValue>
            <hl7:InstanceIdentifier xsi:type="hl7:II" root="2.16.756.5.30.1.127.3.10.3" extension="8901" />
          </xacml-context:AttributeValue>
        </xacml-context:Attribute>
      </xacml-context:Resource>
      <xacml-context:Action />
      <xacml-context:Environment />
    </xacml-context:Request>
  </xacml-samlp:XACMLPolicyQuery
</soap:Body>
```

```
<soap:Body>
  <xacml-samlp:XACMLPolicyQuery>
    <xacml:PolicySetIdReference>urn:e-health-suisse:2015:policies:exclusion-list</xacml:PolicySetIdReference>
  </xacml-samlp:XACMLPolicyQuery>
</soap:Body>
```

Listing 20: Structure of SOAP body element of a XACMLPolicyQuery payload with the xacml-context-syntax to find all patient privacy policies of a specific patient, identified by the patient id. The second example shows a XACMLPolicyQuery by PolicySetId

3.3.8 Expected Actions

All policies satisfying the Resource definitions within a Request SHALL be returned if allowed by ADR. Possible PolicyIdReference or PolicySetIdReference references within the PolicySet(s) to be returned SHALL not be resolved and returned.

3.3.9 XACMLPolicyQuery Response

The XACMLPolicyQuery Response message is created by the Policy Repository in response to the XACMLPolicyQuery Request. In conformance to SAML 2.0 profile of XACML v2.0, the Policy Repository SHALL produce a SAML Assertion response message that conveys the resulting Policies and Policy Sets within a Policy Statement.

3.3.10 Trigger Events

This message is created by the Policy Repository after the evaluation of a XACMLPolicyQuery Request message. The Policy Repository identifies Policy Sets applicable to be returned to the requester.

3.3.11 Message Semantics

The **XACMLPolicy <Assertion>** as specified in OASIS SAML 2.0 Profile of XACML v2.0 (Chapter 5.6), is conveyed within a XACMLPolicy **<Response>**. The Assertion does not need to be signed. The **<Issuer>** of the Assertion MUST identify the Policy Repository. For the EPR this ID is specified to be the home community ID of the Authorization Decision Provider community encoded as an URN, e.g.
<saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.99</saml:Issuer>. The SAML **StatusCode** of the /Assertion/Status of the Response SHALL be conveyed as defined in OASIS SAML 2.0 Profile of XACML v2.0.

```
<soap:Body>
  <samlp:Response ID="4v7a68d0-5d67-557e-def4-8e5858676abc2" Version="2.0"
                  IssueInstant=" 2016-02-09T09:30:10.5Z ">
    <saml:Assertion ID="3b5a66d0-5d86-477e-afc4-8e561084edc9" Version="2.0"
                  IssueInstant=" 2016-02-09T09:30:10.5Z ">
      <saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.999.1</saml:Issuer>
      <saml:Status>
        <samlp:StatusCode>urn:oasis:names:tc:SAML:2.0:status:Success</samlp:StatusCode>
      </saml:Status>
      <saml:Statement xsi:type="xacml-saml:XACMLPolicyStatementType">
        <!—XACML Policy-->
      </saml:Statement>
    </saml:Assertion>
  </samlp:Response>
</soap:Body>
```

Listing 21: Structure of the SOAP body element of the response to a XACMLPolicyQuery. A placeholder is used for the XACML policies returned by the Policy Repository.

3.3.12 EPR AddPolicyRequest and EPR UpdatePolicyRequest

This message enables the Policy Manager to add or update XACML policies, respectively existing XACML Policy Sets of a patient.
This message relies on SAML 2.0 Profile of XACML v2.0.
Actors involved support XUA and use SAML identity assertions to identify current user entities for subsequent access enforcement.

3.3.13 Trigger Events

The Policy Manager sends these messages when it needs to add new or update existing patient- specific policy sets stored within the Policy Repository (of a patient's referenceCommunity).

3.3.14 Message Semantics

This message relies on an EPR specific transaction schema (epd-policy-administration-combined-schema-1.0-local.xsd) as the SAML 2.0 profile of XACML v2.0 does not provide a transaction type and schema REQUIRED by these requests. It uses the element **<AddPolicyRequest>** or **<UpdatePolicyRequest>** to identify the transaction and convey the request.
Otherwise it relies on the very same specification and concepts as the XACMLPolicyQuery Response message does. XACML Policies or Policy Sets to be added or updated are conveyed using a SAML **<Statement>** of type **XACMLPolicyStatementType** within a XACML Policy SAML **<Assertion>** as specified in OASIS SAML 2.0 profile of XACML v2.0. The Assertion does not need to be signed. The **<Issuer>** of the Assertion SHALL identify the Policy Manager. For the EPR this ID is specified to be the home community ID of the Authorization Decision Provider community encoded as an URN, e.g. **<saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.98</saml:Issuer>.**

```
<soap:Body>
  <epd:AddPolicyRequest> <!--or-->
  <epd:UpdatePolicyRequest>
    <saml:Assertion ID="_3b5a66d0-5d86-477e-afc4-8e561084edc9" Version="2.0“
                    IssueInstant="2016-02-09T09:30:10.5Z">
      <saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.999</saml:Issuer>
      <saml:Statement xsi:type="xacml-saml:XACMLPolicyStatementType">
        <!—XACML Policy-->
      </saml:Statement>
    </saml:Assertion>
  </epd:AddPolicyRequest> <!--or-->
  </epd:UpdatePolicyRequest>
</soap:Body>
```

Listing 22: Structure of the SOAP body element of the response to an AddPolicyRequest, with the policy to be conveyed injected in the Statement as denoted by the placeholder.

3.3.15 Expected Actions

The Policy Repository SHALL return a status according to the success or failure of the transaction as defined below.

3.3.16 EPR AddPolicyRequest Response and EPR UpdatePolicyRequest Response

The EPR AddPolicyRequest Response or EPR UpdatePolicyRequest Response message is created by the Policy Repository in response to the EPR AddPolicyRequest or EPR UpdatePolicyRequest message.
An EPR specific transaction EPR PolicyRepositoryResponse is applied to report a general success or failure code. A soap fault MUST be reported back to the Policy Manager in case an EPR UpdatePolicyRequest cannot be executed due to unknown policy or policy set IDs. There is no partial success defined. If at least one policy within the update request cannot be added or updated, the entire request MUST result in a failure/fault response.

3.3.17 Trigger Events

This message is created by the Policy Repository after the EPR AddPolicyRequest or EPR UpdatePolicyRequest have been executed or refused to be executed.

3.3.18 Message Semantics

The EPR specific transaction **<PolicyRepositoryResponse>** conveys a status **urn:e-health-suisse:2015:response-status:success** or **urn:e-health-suisse:2015:response-status:failure**.

```
<soap:Body>
  <epd:EpdPolicyRepositoryResponse status="urn:e-health-suisse:2015:response-status:success"/>
</soap:Body>

<soap:Body>
  <epd:EpdPolicyRepositoryResponse status="urn:e-health-suisse:2015:response-status:failure"/>
</soap:Body>
```

Listing 23: Status element of the response to a request to add or to update a policy.

In case of an update failure due to unknown Policy Set IDs a soap **<Fault>** with a **<Detail> <epd-policy-administration:UnknownPolicySetId>** MUST be returned to the Policy Manager.

```
<soap:Fault>
  <soap:Code>
    <soap:Value>soap:Receiver</soap:Value>
  </soap:Code>
  <soap:Reason>
    <soap:Text xml:lang="en">The PolicySet with the given PolicySet ID does not exist</soap:Text>
  </soap:Reason>
  <soap:Detail>
    <ppq-policy-administration:UnknownPolicySetId xmlns:ppq-policy-administration="urn:e-health-
suisse:2015:policy-administration" />
  </soap:Detail>
</soap:Fault>
```

Listing 24: The soap fault element with error message in the case of an failure of the update request.

3.3.19 EPR DeletePolicyRequest

This message enables the Policy Manager to delete XACML Policies or Policy Sets from a Policy Repository.
This message relies on SAML 2.0 profile of XACML v2.0.
Actors involved support XUA and use SAML identity assertions to identify current user entities for subsequent access enforcement (see ITI TF-1: 39.5 and 39.6).

3.3.20 Trigger Events

The Policy Manager sends these messages when it needs to delete existing patient-specific policy sets stored within the Policy Repository (of a patient's referenceCommunity).

3.3.21 Message Semantics

This message relies on an EPR specific transaction schema (epd-policy-administration-combined-schema-1.0-local.xsd) as the SAML 2.0 profile of XACML does not provide a transaction type and schema REQUIRED by this requests. It uses the element **<DeletePolicyRequest>** to identify the transaction and convey the request.
Otherwise it relies on the same specification and concepts as the XACMLPolicyQuery Response message, EPD AddPolicyRequest and EPR UpdatePolicyRequest do. However, there is no Statement type specified to convey the information needed by this transaction. Policies or Policy Sets to be deleted are to be identified by a corresponding ID that is to be conveyed using an EPR specific SAML **<Statement>** of type **XACMLPolicySetIdReferenceStatementType** (as defined in epd-policy-administration-combined-schema-1.0-local.xsd) within a XACML Policy SAML **<Assertion>**. The Assertion does not need to be signed. The ID of a deleted Policy MUST not be reused.
The **<Issuer>** of the Assertion SHALL identify the Policy Manager. For the EPR this ID is specified to be the home community ID of the Authorization Decision Provider community encoded as an URN, e.g.
<saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.98</saml:Issuer>.

```
<soap:Body>
  <epd:DeletePolicyRequest>
    <saml:Assertion ID="_3b5a66d0-5d86-477e-afc4-8e561084edc9" Version="2.0“
                    IssueInstant="2016-02-09T09:30:10.5Z">
      <saml:Issuer NameQualifier="urn:e-health-suisse:community-index">urn:oid:2.999</saml:Issuer>
      <saml:Statement xsi:type="epd:XACMLPolicySetIdReferenceStatementType">
        <xacml:PolicySetIdReference>10a3f268-d9d6-4772-b908-9d8521161</xacml:PolicySetIdReference>
      </saml:Statement>
    </saml:Assertion>
  </epd:DeletePolicyRequest>
</soap:Body>
```

Listing 25: Structure of the SOAP body for a EPR DeletePolicyRequest, where the policy set to be removed from the patients policy configuration is referenced by ID.

3.3.22 Expected Actions

The Policy Repository SHALL return a status according to the success or failure of the transaction as defined below.

3.3.23 EPR DeletePolicyRequest Response

The EPR DeletePolicyRequest Response message is created by the Policy Repository in response to the EPR DeletePolicyRequest.
An EPR specific transaction EPR PolicyRepositoryResponse is applied to report a general success or failure code. A soap fault MUST be reported back to the Policy Manager in case an EPR DeletePolicyRequest cannot be executed due to unknown Policy or Policy Set IDs.

3.3.24 Trigger Events

This message is created by the Policy Repository after the EPR DeletePolicyRequest or have been executed or refused to be executed.

3.3.25 Message Semantics

The EPR specific transaction **<PolicyRepositoryResponse>** conveys the status **urn:e-health-suisse:2015:response-status:success** or **urn:e-health-suisse:2015:response-status:failure**.

```
<soap:Body>
  <epd:EpdPolicyRepositoryResponse status="urn:e-health-suisse:2015:response-status:success"/>
</soap:Body>

<soap:Body>
  <epd:EpdPolicyRepositoryResponse status="urn:e-health-suisse:2015:response-status:failure"/>
</soap:Body>
```

Listing 26: Status element of the response to a request to delete a policy.

In case of a failure of the delete request due to unknown Policy Set IDs a soap **<Fault>** with a **<Detail> <epd-policy-administration:UnknownPolicySetId/>** MUST be returned to the Policy Manager.

```
<soap:Fault>
   <soap:Code>
      <soap:Value>soap:Receiver</soap:Value>
   </soap:Code>
   <soap:Reason>
      <soap:Text xml:lang="en">The PolicySet with the given PolicySet ID does not exist</soap:Text>
   </soap:Reason>
   <soap:Detail>
      <ppq-policy-administration:UnknownPolicySetId xmlns:ppq-policy-administration="urn:e-health-
suisse:2015:policy-administration" />
   </soap:Detail>
</soap:Fault>
```

Listing 27: The soap fault element with error message in the case of a failure of the delete request.

3.3.26 Security Considerations

Relevant Security Considerations are defined in ITI TF-1: 39.5. The Privacy Policy Query transactions require TLS communication between actors involved. Relevant XDS Affinity Domain Security background is discussed in the XDS Security Considerations Section (see ITI TF-1: 10.7). The Actors involved SHALL record audit events according to the following:

3.3.27 Policy Manager Audit Message

	Field Name	**Opt**	**Value Constraints**
Event	EventID	M	EV (110112, DCM,"Query")
	EventActionCode	M	E = Execute
	EventDateTime	*M*	*not specialized*
	EventOutcomeIndicator	*M*	*not specialized*
	EventTypeCode	M	EV("PPQ", "e-health-suisse", "Privacy Policy Query Policy Query") *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Add Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Update Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Delete Policy")*
Source (Policy Manager) (1)			
Human Requestor (0..n)			
Destination (Document Registry) (1)			
Audit Source (Document Consumer) (1)			
Patient (0..1)			
Query Parameters(1..n)			

Source: AuditMessage/ ActiveParticipant	*UserID*	*U*	*not specialized*
	AlternativeUserID	M	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110153, DCM, "Source")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address.

Human Requestor (if known) AuditMessage/ ActiveParticipant	UserID	M	Identity of the human that initiated the transaction.
	AlternativeUserID	*U*	*not specialized*
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	U	Access Control role(s) the user holds that allows this transaction.
	NetworkAccessPointTypeCode	NA	
	NetworkAccessPointID	NA	

Destination AuditMessage/ ActiveParticipant	UserID	M	SOAP endpoint URI.
	AlternativeUserID	U	not specialized
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	U	EV(110152, DCM, "Destination")
	NetworkAccessPointTypeCode	M	"1" for machine (DNS) name, "2" for IP address
	NetworkAccessPointID	M	The machine name or IP address.

Audit Source AuditMessage/ AuditSourceIdentification	*AlternativeUserID*	*U*	*not specialized*
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*

Patient (AuditMessage/ ParticipantObjectIdentification)	ParticipantObjectTypeCode	M	"1" (person)
	ParticipantObjectTypeCodeRole	M	"11" (patient)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	*not specialized*
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	The patient ID in HL7 CX format.
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	*U*	*not specialized*

Query Parameters: AuditMessage/ ParticipantObjectIdentification (1..n)	ParticipantObjectTypeCode	M	"2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"24" (query)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("PPQ", "e-health-suisse", "Privacy Policy Query Policy Query") *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Add Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Update Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Delete Policy")*
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	*M*	*not specialized*
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	M	For Add, Update, Delete transaction: PolicySetId(s) - If multiple Policies are present, only one ATNA is present with a list of identifiers separated by comma; For query all policies of a patient transaction: PatientId
	ParticipantObjectDetail	*U*	*not specialized*

3.3.28 Policy Repository Audit Message

	Field Name	Opt	Value Constraints
Event	EventID	M	EV (110112, DCM,"Query")
	EventActionCode	M	E = Execute
	EventDateTime	*M*	*not specialized*
	EventOutcomeIndicator	*M*	*not specialized*
	EventTypeCode	M	EV("PPQ", "e-health-suisse", "Privacy Policy Query Policy Query") *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Add Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Update Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Delete Policy")*
Source (Policy Manager) (1)			
Destination (Policy Repository) (1)			
Audit Source (Policy Repository) (1)			
Patient (0..1)			
Query Parameters (1..n)			

Source: AuditMessage/ ActiveParticipant	UserID	*M*	*not specialized*
	AlternativeUserID	U	*not specialized*
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110153, DCM, "Source")
	NetworkAccessPointTypeCode	M	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address.

Destination: AuditMessage/ ActiveParticipant	UserID	M	SOAP endpoint URI.
	AlternativeUserID	M	the process ID as used within the local operating system in the local system of logs
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*
	RoleIDCode	M	EV (110152, DCM, "Destination")
	NetworkAccessPointTypeCode	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address.

Audit Source AuditMessage/ AuditSourceIdentification	*AlternativeUserID*	*U*	*not specialized*
	UserName	*U*	*not specialized*
	UserIsRequestor	*U*	*not specialized*

Patient (AuditMessage/ ParticipantObjectIdentification)	ParticipantObjectTypeCode	M	"1" (person)
	ParticipantObjectTypeCodeRole	M	"1" (patient)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	*not specialized*
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	M	The patient ID in HL7 CX format.
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	*U*	*not specialized*
	ParticipantObjectDetail	*U*	*not specialized*

Query Parameters: AuditMessage/ ParticipantObjectIdentification	ParticipantObjectTypeCode	M	"2" (SYSTEM)
	ParticipantObjectTypeCodeRole	M	"24" (query)
	ParticipantObjectDataLifeCycle	*U*	*not specialized*
	ParticipantObjectIDTypeCode	M	EV("PPQ", "e-health-suisse", "Privacy Policy Query Policy Query") *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Add Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Update Policy")* *EV*("PPQ", "e-health-suisse", *"Privacy Policy Query Delete Policy")*
	ParticipantObjectSensitivity	*U*	*not specialized*
	ParticipantObjectID	*M*	*not specialized*
	ParticipantObjectName	*U*	*not specialized*
	ParticipantObjectQuery	M	For Add, Update, Delete transaction: PolicySetId(s) - If multiple Policies are present, only one ATNA is present with a list of identifiers separated by comma; For query all policies of a patient transaction: PatientId
	ParticipantObjectDetail	*U*	*not specialized*

4 Volume 3 – Content Profiles

4.1 Privacy Policy Format

4.1.1 Scope

Following the Swiss regulations, any patient has the right to manage the access rights to his personal EPR. The access rights are implemented using the OASIS Policy Schema of the XACML 2.0 Specification Set with the use of the HL7 V3 Data types 'Coded Value' and 'Instance Identifier'.
Detailed contents to be provided in the Privacy Policy Format are described in the following chapters.

For the correct enforcement of the access rights formulated by the patient, the following definition is significant:

1. **Read** enforcement decisions base on a **maximum** access level definied by the patient in his privacy policy.
2. **Write** enforcement decisions base on a **minimum** provide level definied by the patient in his privacy policy.

Legend:
✓: Indicates an CH:ADR transaction response that contains a permit.
×: Indicates an CH:ADR transaction response that contains a deny.

Table 3: EPR access level matrix

Read EPR		**Confidentiality Level**			**Patients setting**	
		normal	restricted	secret		
EPR User	Health Care Professional (real person granted by the Patient)	✓	×	×	normal	**Maximal Access Level**
		✓	✓	×	restricted	
		✓[2]	×[3]	×	emergency	
		×	×	×	exclusion-list[4]	
	Patient	✓	✓	✓	full	

[2] The patient can change this behavior using the Privacy Manager in order to let CH:ADR transaction responses deny: ×

[3] The patient can change this behavior using the Privacy Manager in order to let CH:ADR transaction responses permit: ✓ (if and only if his choice for Confidentiality Level 'normal' is as well ✓!)

[4] The exclusion list overrides all other settings: If a health care professional is on a patient's exclusion-list, all read enforcement decisions are always deny!

Table 4: EPR provide level matrix

Write EPR		**Confidentiality Level**[5]			**Patients setting**[6]	
		normal	restricted	secret		
EPR Role	Health Care Professional (role of the user; impersonal)	✓	✓	×	normal	**Minimal Provide Level**
		×	✓	×	restricted	
	Patient	✓	✓	✓	full	

The Privacy Policy Format consists of two element types:

- Static policies valid for all communities within the EPR circle of trust.
  See 4.1.1.1 EPR Policy Stack.
- Dynamic policy sets for a patient specific EPR.
  See 4.1.1.2 Patient Specific Policy Sets.

4.1.1.1 EPR Policy Stack

Base Policies and Access Level Policy Sets are static elements and therefore identical within the Swiss EPR circle of trust. They MUST be implemented once in the Policy Repository of each referenceCommunity. See https://www.bag.admin.ch/epra for the normative XACML representations.

4.1.1.1.1 Base Policies

The Base Policies describe in general the permissions for access (read) or provide (write) information in an EPR with respect to the level of confidentiality that is assigned to the information (read/write - normal/restricted/secret) .
Other Base Policies describe the access to the policy repository (policyadmin) or explicitly prohibit the access to an EPR and the policy repositories (deny all) .

4.1.1.1.2 Access Level Policy Sets

The Access Level Policy Sets describe the access levels (normal, restricted, full, exclusion-list) and provide levels (normal, restricted) . They refer to one or more Base Policy according to their characteristics.

The access levels define the maximum confidentiality level that will be permitted while reading from an EPR. E.g. a read request to a 'restricted' document will be denied for access level 'normal' and a read request to a 'normal' document will be permitted for access level 'restricted'.

The provide levels define the minimum confidentiality level that will be permitted while writing to an EPR. E.g. a write request of a 'normal' document will be denied for provide level 'restricted' and a write request to a 'restricted' document will be permitted for provide level 'normal'.

Access Level Policy Sets that describe delegation rights in addition to the access level (normal, restricted), contain themselves polices according to their characteristics, which allows access to policy repositories (AddPolicy/PolicyQuery)

[5] Creation of new documents and update Metadata for existing documents are basically both handled using the provide level matrix. There is one exception: All write request enforcement decisions for updates of confidentiality codes are always deny for health professionaly. Only the patient is allowed to update confidentiality codes. This has to be handled by the Authorization Decision Provider and does not affect the Privacy Policy.

[6] These settings are independent of the purpose of use (normal access / emergency access)

The PolicySetIds are fixed names as described in the following chapters.

4.1.1.2 Patient Specific Policy Sets

Default Patient Specific Policy Sets are created during the creation of a patient's EPR. They might be modified by the patient via Policy Manager. While the patient can refine the access rights via Policy Manager, there might also other Patient Specific Policy Sets be created, or existing Patient Specific Policy Sets might be deleted.

The PolicySetIds are UUIDs, which are to be generated by the Policy Manager.
See https://www.bag.admin.ch/epra for the informative XACML representations.

4.1.1.2.1 EPR Setup

When creating an EPR for a patient, his Patient Specific Policy Sets are created and stored in the Policy Repository. The patient is explicitly granted full access (reference to full access level). The patient (respectively his representative) is the only one who can access secret data.
In addition, the default access rule for emergency access is created by referring to the default option of the emergency access level. Furthermore, the default requirement for the minimal confidentiality level that will be permitted while writing to an EPR is created. The references to the emergency access level or the minimal confidentiality level that will be permitted while writing to an EPR can be modified by the patient via Policy Manager.

The initial creation of the Policy Sets during the EPR Setup must take place in the reference community of the patient using a privileged administration access to the Policy Repository!

The only change allowed in the informative XACML representations are:
1. PolicySetId: UUID generated by the Policy Manager
2. EPR-PID in the SubjectMatch
3. PolicySetIdReference: can changed later only by Patient using the Policy Manager.
   Must remain urn:e-health-suisse:2015:policies:access-level:normal during EPR Setup!

4.1.1.2.2 User Assignment

The assignment of health care professionals or groups of health care professionals to a specific access level is done by creating Policy Sets for each user or group to be authorized.
Using the same mechanism, a patient can grant any other user or group within the EPR circle of trust for the administration (create, modify, delete) of his access rights (Policy Sets). These rights are applied by the Policy Manager.

The informative XACML representation shows an example, only.

4.1.2 Referenced Standards

- OASIS eXtensible Access Control Markup Language (XACML) Version 2.0 XACML Core 2.0
  http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf
- OASIS Policy Schema of the XACML 2.0 Specification Set
  Schema: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-policy-schema-os.xsd
- HL7v3 Abstract Data Type Specification - ANSI/HL7 V3 DT, R1-2004 3/19/2012
  https://www.hl7.org/implement/standards/product_brief.cfm?product_id=362
  Schema: ftp://ftp.ihe.net/TF_Implementation_Material/ITI/schema/IHE/APPC/ihe-appc-xacml-hl7-datatypes-base-1.0.xsd

4.1.3 Detailed Privacy Policy Format definitions

The detailed specifications for the Policy requirements specified within the XACML 2.0 Specification Set MUST be used with the following Swiss precisions.

4.1.3.1 Policy

Refers to data type xacml:PolicyType

Table 5: xacml:PolicyType definitions

Element Name	Card.	Swiss Precision
attributes:		
PolicyId	1..1	See Value-Set in section 4.1.4.1 PolicyId
Version	0..1	Not permitted
RuleCombiningAlgId	1..1	urn:oasis:names:tc:xacml:1.0:rule-combining-algo-rithm:deny-overrides
Description	0..1	See https://www.bag.admin.ch/epra
PolicyDefaults	0..1	Not permitted
CombinerParameters	0..1	Not permitted
Target	1..1	Must be declared as empty element for embedded Policies in Base Policy Sets for delegation. For Base Policies, see Table 6: xacml:TargetType definitions for Policy.
choice:	1..*	1..1
CombinerParameters	0..1	Not permitted
RuleCombinerParameters	0..1	Not permitted
VariableDefinition	1..1	Not permitted in this choice.
Rule	1..1	See Table 7: xacml:RuleType definitions.
Obligations	0..1	Not permitted

Table 6: xacml:TargetType definitions for Policy

Element Name	Card.	Swiss Precision
Subjects	0..1	Not permitted
Resources	0..1	1..1
Resource	1..*	No further refinement
ResourceMatch	1..*	See Table 8: xacml:ResourceMatchType definitions for Policies.
Actions	0..1	1..1
Action	1..*	No further refinement
ActionMatch	1..*	See Table 11: xacml:ActionMatchType definitions.
Environments	0..1	Not permitted

Table 7: xacml:RuleType definitions

Element Name	Card.	Swiss Precision
attributes:		
RuleId	1..1	GUID defined by the Policy Manager
Effect	1..1	See Value-Set in section 4.1.4.3 Rule Effect
Description	0..1	Not permitted
Target	0..1	Not permitted
Condition	0..1	Not permitted for Base Policies [0..0]. Required [1..1] for embedded Policies in Base Policy Sets for delegation, see Table 14: xacml:ConditionType definitions for embedded Policies in Base Policy Sets for delegation.

Table 8: xacml:ResourceMatchType definitions for Policies

Element Name	Card.	Swiss Precision
attributes:		
MatchId	1..1	urn:hl7-org:v3:function:CV-equal
AttributeValue	1..1	See Table 9: xacml:AttributeValueType definitions for ResourceMatch for Policies.
choice:	1..1	No further refinement
ResourceAttributeDesignator	1..1	See Table 12: xacml:AttributeDesignatorType definitions for ResourceMatch for Policies.
AttributeSelector	1..1	Not permitted in this choice.

Table 9: xacml:AttributeValueType definitions for ResourceMatch for Policies

Element Name	Card.	Swiss Precision
attributes:		
xs:anyAttribute	0..1	[1..1] DataType: urn:hl7-org:v3#CV
xs:any	0..1	[1..1] hl7:CodedValue codeSystem="2.16.840.1.113883.6.96" code: Valid value according to Swiss Metadata, table Confidentiality level

Table 10: xacml:AttributeValueType definitions for ActionMatch

Element Name	Card.	Swiss Precision
attributes:		
xs:anyAttribute	0..1	[1..1] DataType: http://www.w3.org/2001/XMLSchema#anyURI
xs:any	0..1	Not permitted
Text	0..1	[1..1] See Value-Set in section 4.1.4.2 Actions

Table 11: xacml:ActionMatchType definitions

Element Name	Card.	Swiss Precision
attributes:		
MatchId	1..1	urn:oasis:names:tc:xacml:1.0:function:anyURI-equal
AttributeValue	0..1	Not permitted
choice:	0..1	Not permitted
ActionAttributeDesignator	0..1	[1..1] See Table 13: xacml:AttributeDesignatorType definitions for ActionMatch
AttributeSelector	0..1	Not permitted

Table 12: xacml:AttributeDesignatorType definitions for ResourceMatch for Policies

Element Name	Card.	Swiss Precision
attributes:		
AttributeId	1..1	urn:ihe:iti:xds-b:2007:confidentiality-code
DataType	1..1	urn:hl7-org:v3#CV
Issuer	0..1	Not permitted
MustBePresent	0..1	Not permitted

Table 13: xacml:AttributeDesignatorType definitions for ActionMatch

Element Name	Card.	Swiss Precision
attributes:		
AttributeId	1..1	urn:oasis:names:tc:xacml:1.0:action:action-id
DataType	1..1	http://www.w3.org/2001/XMLSchema#anyURI
Issuer	0..1	Not permitted
MustBePresent	0..1	Not permitted

Table 14: xacml:ConditionType definitions for embedded Policies in Base Policy Sets for delegation

Element Name	Card.	Swiss Precision
Apply	1..1	Ensure that there is the correct access level policySet referenced
attributes:		
FunctionId	1..1	urn:oasis:names:tc:xacml:2.0:function:anyURI-regexp-match
AttributeValue	1..1	No further refinement
attributes:		
DataType	1..1	http://www.w3.org/2001/XMLSchema#string
text	1..1	For normal access levels: (urn:e-health-suisse:2015:policies:access-level:)(normal).* For normal or restricted access levels: (urn:e-health-suisse:2015:policies:access-level:)(normal|restricted).*
Apply	1..1	Ensure that there is one policySet referenced
attributes:		
FunctionId	1..1	urn:oasis:names:tc:xacml:1.0:function:anyURI-one-and-only
ResourceAttributeDesignator	1..1	No further refinement
attributes:		
DataType	1..1	http://www.w3.org/2001/XMLSchema#anyURI
AttributeId	1..1	urn:e-health-suisse:2015:policy-attributes:referenced-policy-set

4.1.3.2 PolicySet

Refers to data type xacml:PolicySetType

Table 15: xacml:PolicySetType definitions

Element Name	Card.	Swiss Precision
attributes:		
PolicySetId	1..1	See Value-Set in section 4.1.4.4 PolicySetId
Version	0..1	Not permitted
PolicyCombiningAlgId	1..1	urn:oasis:names:tc:xacml:1.0:policy-combining-algorithm:deny-overrides
Description	0..1	See https://www.bag.admin.ch/epra
PolicySetDefaults	0..1	Not permitted
Target	1..1	See Table 16: xacml:TargetType definitions for *PolicySet*
choice:	0..*	No further refinement
PolicySet	1..1	Not permitted in this choice.
Policy	1..1	Required for embedded Policies in Base Policy Sets for delegation, see section 4.1.3.1 Policy. Not permitted for other PolicySets.
PolicySetIdReference	1..1	See Table 17: xacml:IdReferenceType definitions
PolicyIdReference	1..1	See Table 18: xacml:IdReferenceType definitions for Policies
CombinerParameters	1..1	Not permitted in this choice.
PolicyCombinerParameters	1..1	Not permitted in this choice.
PolicySetCombinerParameters	1..1	Not permitted in this choice.
Obligations	0..1	Not permitted

Table 16: xacml:TargetType definitions for PolicySet

Element Name	Card.	Swiss Precision
Subjects	0..1	Required [1..1] for EPR Setup Policy Sets (except for write permissions) and User Assignment Policy Sets. Not permitted for other PolicySets [0..0].
Subject	1..*	No further refinement
SubjectMatch	1..*	Required for EPR Setup Policy Sets (except for write permissions) and User Assignment Policy Sets, see Table 19: xacml:SubjectMatchType definitions.
Resources	0..1	Required [1..1] for EPR Setup Policy Sets and User Assignment Policy Sets. Not permitted for other PolicySets [0..0].
Resource	1..*	No further refinement
ResourceMatch	1..*	Required for EPR Setup Policy Sets and User Assignment Policy Sets, see Table 20: xacml:ResourceMatchType definitions for PolicySets.
Actions	0..1	Not permitted
Environments	0..1	Required [1..1] for User Assignment PolicySets. Not permitted for other PolicySets [0..0].
Environment	1..*	No further refinement
EnvironmentMatch	1..*	See Table 23: xacml:EnvironmentMatchType definitions.

Table 17: xacml:IdReferenceType definitions for PolicySets

Element Name	Card.	Swiss Precision
attributes:		
Version	0..1	Not permitted
EarliestVersion	0..1	Not permitted
LatestVersion	0..1	Not permitted
text	1..1	See Value-Set in section 4.1.4.4 PolicySetId

Table 18: xacml:IdReferenceType definitions for Policies

Element Name	Card.	Swiss Precision
attributes:		
Version	0..1	Not permitted
EarliestVersion	0..1	Not permitted
LatestVersion	0..1	Not permitted
text	1..1	See Value-Set in section 4.1.4.1 PolicyId

Table 19: xacml:SubjectMatchType definitions

Element Name	Card.	Swiss Precision
attributes:		
MatchId	1..1	For GLN of health professionals: urn:oasis:names:tc:xacml:1.0:function:string-equal According to the EPR SAML 2.0 Assertions Subject/@NameQualifier attribute: urn:oasis:names:tc:xacml:1.0:function:anyURI-equal For EPR-PID of patients: urn:hl7-org:v3:function:II-match For user roles and purpose of use: urn:hl7-org:v3:function:CV-equal
AttributeValue	1..1	No further refinement.
attributes:		
xs:anyAttribute:	0..1	[1..1] For GLN of health professionals: • For the GLN: DataType: http://www.w3.org/2001/XMLSchema#string • For the URI: DataType: http://www.w3.org/2001/XMLSchema#anyURI For EPR-PID of patients: DataType: urn:hl7-org:v3#II For user roles: DataType: urn:hl7-org:v3#CV For purpose of use: DataType: urn:hl7-org:v3#CV

Element Name	Card.	Swiss Precision
xs:any	0..1	[1..1] For GLN of health professionals: DataType: Not permitted For EPR-PID of patients: hl7:InstanceIdentifier @root: 2.16.756.5.30.1.127.3.10.3 @extension: The real EPR-PID of the patient For user roles: hl7:CodedValue @code: PAT or HCP @codeSystem: 2.16.756.5.30.1.127.3.10.4 For purpose of use: hl7:CodedValue @code: NORM or EMER @codeSystem: 2.16.756.5.30.1.127.3.10.5
text	0..1	For GLN of health professionals [1..1]: • For the GLN: DataType: The GLN of the health professional • For the URI: DataTape: urn:gs1:gln For EPR-PID of patients [0..0]: Not permitted For user roles [0..0]: Not permitted For purpose of use [0..0]: Not permitted
choice:	1..1	No further refinement
SubjectAttributeDesignator	1..1	No further refinement
attributes:		
AttributeId	1..1	For GLN of health professionals: • For the GLN: urn:oasis:names:tc:xacml:1.0:subject:subject-id • For the URI: urn:oasis:names:tc:xacml:1.0:subject:subject-id-qualifier For EPR-PID of patients: urn:oasis:names:tc:xacml:1.0:subject:subject-id For user roles: urn:oasis:names:tc:xacml:2.0:subject:role For purpose of use: urn:oasis:names:tc:xspa:1.0:subject:purposeofuse
DataType	1..1	For GLN of health professionals: • For the GLN: http://www.w3.org/2001/XMLSchema#string • For the URI: http://www.w3.org/2001/XMLSchema#anyURI For EPR-PID of patients: http://www.w3.org/2001/XMLSchema#string For user roles: urn:hl7-org:v3#CV For purpose of use: urn:hl7-org:v3#CV

Element Name	Card.	Swiss Precision
Issuer	0..1	Not permitted
MustBePresent	0..1	Not permitted
SubjectCategory	0..1	Not permitted
AttributeSelector	1..1	Not permitted in this choice.

Table 20: xacml:ResourceMatchType definitions for PolicySets

Element Name	Card.	Swiss Precision
attributes:		
MatchId	1..1	urn:hl7-org:v3:function:II-equal
AttributeValue	1..1	See Table 21: xacml:AttributeValueType definitions for ResourceMatch for PolicySets
choice:	1..1	No further refinement
ResourceAttributeDesignator	1..1	See Table 22: xacml:AttributeDesignatorType definitions for ResourceMatch for PolicySets
AttributeSelector	1..1	Not permitted in this choice.

Table 21: xacml:AttributeValueType definitions for ResourceMatch for PolicySets

Element Name	Card.	Swiss Precision
attributes:		
xs:anyAttribute	0..1	[1..1] DataType: urn:hl7-org:v3:function:II-equal
xs:any	0..1	[1..1] hl7:InstanceIdentifier @root: 2.16.756.5.30.1.127.3.10.3 @extension: The EPR-PID of the patient

Table 22: xacml:AttributeDesignatorType definitions for ResourceMatch for PolicySets

Element Name	Card.	Swiss Precision
attributes:		
AttributeId	1..1	urn:e-health-suisse:2015:epd-pid
DataType	1..1	urn:hl7-org:v3#II
Issuer	0..1	Not permitted
MustBePresent	0..1	Not permitted

Table 23: xacml:EnvironmentMatchType definitions

Element Name	Card.	Swiss Precision
attributes:		
MatchId	1..1	For Valid from date: urn:oasis:names:tc:xacml:1.0:function:date-greater-than-or-equal For Valid to date: urn:oasis:names:tc:xacml:1.0:function:date-less-than-or-equal
AttributeValue	0..1	[1..1] See Table 24: xacml:AttributeValueType definitions for EnvironMatch for PoliciySets
choice:	0..1	No further refinement
EnvironmentAttributeDesignator	0..1	[1..1] See Table 25: xacml:AttributeDesignatorType definitions for EnvironmentAttributeDesignator
AttributeSelector	0..1	Not permitted

Table 24: xacml:AttributeValueType definitions for EnvironMatch for PoliciySets

Element Name	Card.	Swiss Precision
attributes:		
xs:anyAttribute	0..1	[1..1] DataType: http://www.w3.org/2001/XMLSchema#date
xs:any	0..1	Not permitted
text	0..1	[1..1] Timestamp reflecting the Valid from / Valid to date Format: According to the W3C XSD Date Data Type. UTC time is required. Timezones are not permitted. Use offset from the UTC time.

Table 25: xacml:AttributeDesignatorType definitions for EnvironmentAttributeDesignator

Element Name	Card.	Swiss Precision
attributes:		
AttributeId	1..1	urn:oasis:names:tc:xacml:1.0:environment:current-date
DataType	1..1	http://www.w3.org/2001/XMLSchema#date
Issuer	0..1	Not permitted
MustBePresent	0..1	Not permitted

4.1.4 Value-Sets

4.1.4.1 PolicyId

The following PolicyId values are permitted:

- urn:e-health-suisse:2015:policies:permit-reading-normal
- urn:e-health-suisse:2015:policies:permit-reading-restricted
- urn:e-health-suisse:2015:policies:permit-reading-secret
- urn:e-health-suisse:2015:policies:permit-writing-normal
- urn:e-health-suisse:2015:policies:permit-writing-restricted
- urn:e-health-suisse:2015:policies:permit-writing-secret
- urn:e-health-suisse:2015:policies:full-policy-administration
- urn:e-health-suisse:2015:policies:deny-all
- urn:e-health-suisse:2015:policies:delegation-up-to-normal

4.1.4.2 Actions

The following Action values are permitted:

- Reading:
  - urn:ihe:iti:2007:RegistryStoredQuery
  - urn:ihe:iti:2007:RetrieveDocumentSet
  - urn:ihe:iti:2007:CrossGatewayQuery
  - urn:ihe:iti:2007:CrossGatewayRetrieve
  - urn:ihe:iti:2011:CrossGatewayFetch
- Writing:
  - urn:ihe:iti:2007:RegisterDocumentSet-b
  - urn:ihe:iti:2007:ProvideAndRegisterDocumentSet-b
- Policy administration:
  - urn:e-health-suisse:2015:policy-administration:PolicyQuery
  - urn:e-health-suisse:2015:policy-administration:AddPolicy
  - urn:e-health-suisse:2015:policy-administration:UpdatePolicy
  - urn:e-health-suisse:2015:policy-administration:DeletePolicy

4.1.4.3 Rule Effect

The following Rule Effect values are permitted:

- Permit
- Deny

4.1.4.4 PolicySetId

The following PolicySetId values are permitted:

- urn:uuid:<UUID> (patient specific policy stack)
- urn:e-health-suisse:2015:policies:access-level:normal
- urn:e-health-suisse:2015:policies:access-level:restricted
- urn:e-health-suisse:2015:policies:access-level:delegation-and-restricted
- urn:e-health-suisse:2015:policies:access-level:delegation-and-normal
- urn:e-health-suisse:2015:policies:access-level:full
- urn:e-health-suisse:2015:policies:provide-level:normal
- urn:e-health-suisse:2015:policies:provide-level:restricted
- urn:e-health-suisse:2015:policies:exclusion-list

5 Appendix

Figures

Tables