Authorisation Infrastructure
for
On-Demand Network Resource Provisioning

Yuri Demchenko
System and Network Engineering Group
University of Amsterdam

Grid2008 Conference, 29 September- 1 October 2008, Tsukuba

Outline

- AAA/AuthZ Architecture for Optical Network Resource Provisioning (NRP)
- AAA/AuthZ mechanisms and functional components to support multidomain NRP
  - AuthZ ticket for extended AuthZ context management
  - Token Validation Service (TVS) and token-based access control and signalling
- XACML-NRP attributes and policy profile
- Reference Model for Obligations Handling (OHRM)
- Future developments
  - Using Identity Based Cryptography (IBC) for building dynamic security associations for provisioned resources

Background for this research

- EU funded Phosphorus Project “Lambda User Controlled Infrastructure for European Research” (EC Contract number 034115)
- University of Amsterdam SNE Group ongoing research on GAAA-AuthZ – Generic Authentication, Authorization, Accounting (GAAA) AuthZ Framework

Optical Network Resource Provisioning (NRP)

NRP as a use case of the general Complex Resource Provisioning (CRP)

- ONRP and Network on-demand provisioning
- Grid Computing Resource – Distributed and heterogeneous

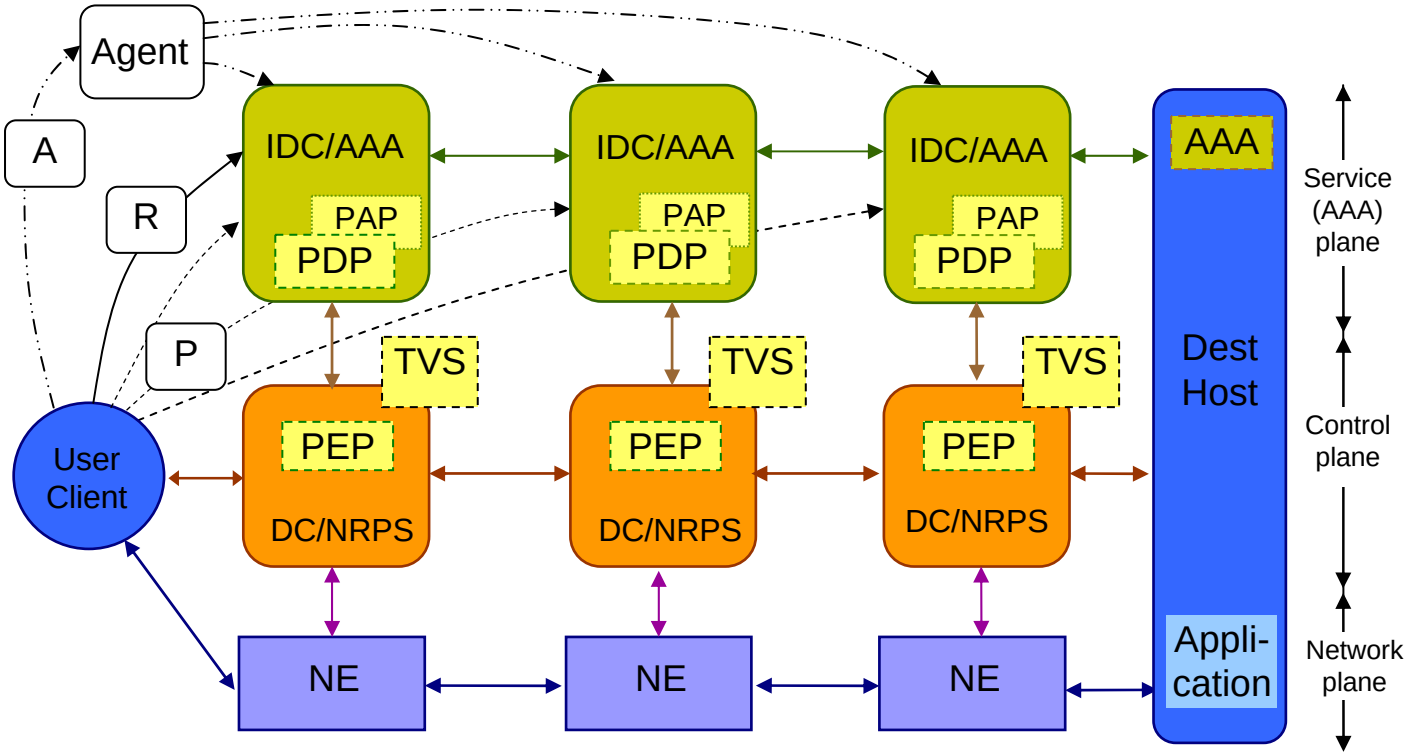
3 major stages/phases in NRP/CRP operation/workflow

- Provisioning consisting of 3 basic steps
  - Resource Lookup
  - Resource composition (including options)
  - Component resources reservation (in advance), including combined AuthZ/policy decision, and assigning a global reservation ID (GRI)
- Deployment – reservation confirmation and distributing components/domain configuration (including trusted keys)
- Access (to the reserved resource) or consumption (of the consumable resource)

Now considering two other stages: “decommissioning” and “relocation”

- Topic for future research and discussions
- Will allows integrating resource provisioning into the upper layer scientific workflow in more consistent way

Multidomain Network Resource Provisioning (NRP)



Provisioning sequences

- Agent (A)
- Polling (P)
- Relay (R)

Token based policy enforcement

GRI – Global Reservation ID

AuthZ tickets for multidomain context mngnt

IDC – Interdomain Controller

DC – Domain Controller

NRPS – Network Resource Provisioning System

AAA – AuthN, AuthZ, Accounting Server
PDP – Policy Decision Point
PEP – Policy Enforcement Point
TVS – Token Validation Service
KGS – Key Generation Service

Multi-domain NRP – Domain definition

Domains are defined (as associations of entities) by a common policy under single administration, common namespaces and semantics, shared trust,etc.

Domain related security context may include

- namespace aware names and ID's
- policy references/ID's
- trust anchors
- authority references
- Additionally, each domain may have/create own dynamic/session related security context (at the reservation and access stages)

Multi-domain NRP AuthZ infrastructure

- Multiple policies processing and combination, including obligated/conditional policy decisions and delegation
- Attributes/rules mapping/converting based on inter domain trust management infrastructure
- Policy support for different logical organisation of resources, including possible constraints on resource combination and interoperation

AAA/AuthZ mechanisms and functional components to support multidomain ONRP

The proposed AAA/security mechanisms and functional components to extend generic AAA AuthZ framework (PEP, PDP, PAP and operational sequences)

Token Validation Service (TVS) to enable token based policy enforcement and signalling

- Can be applied at all Networking layers (Service, Control and Data planes)

AuthZ ticket format for extended AuthZ session management

- To allow extended AuthZ decision/security context communication between domains

XACML attributes and policy profile for NRP

- Using reach functionality of the XACML policy format for controlling complex network and Grid resources
- *Can add dynamic path/topology information to policy definition*

Policy Obligation Handling Reference Model (OHRM)

- Used for account mapping, quota enforcement, accounting, etc.

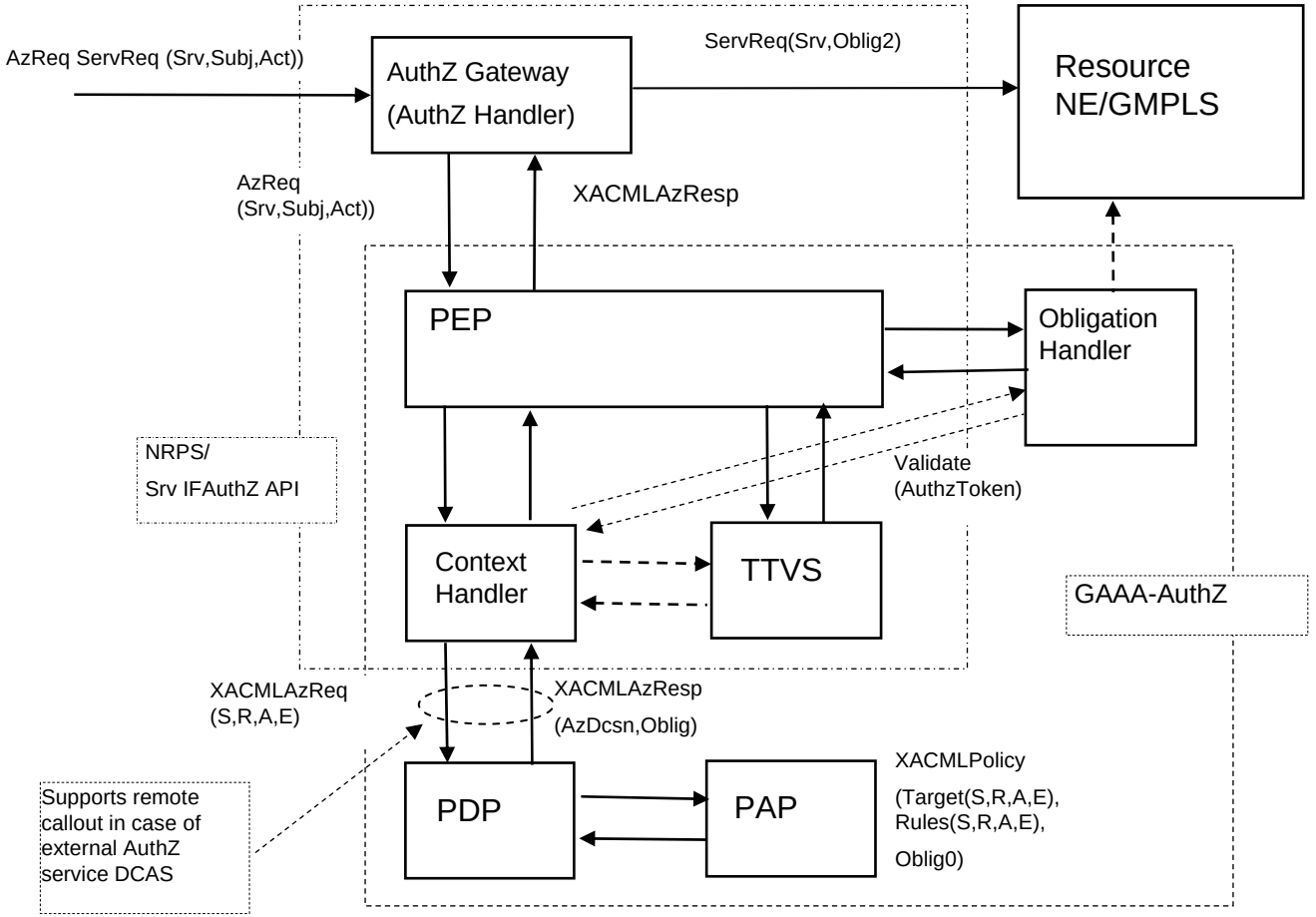
Identity Based Cryptography (IBC) use for token key distribution in inter-domain network resource provisioning is being investigated

- Targeted for the “deployment” stage

The proposed architecture will allow smooth integration with other AuthZ frameworks as currently used and being developed by NREN and Grid community

- Can provide basic AAA/AuthZ functionality for each network layer DP, CP, SP

GAAA Toolkit pluggable AAA/AuthZ components

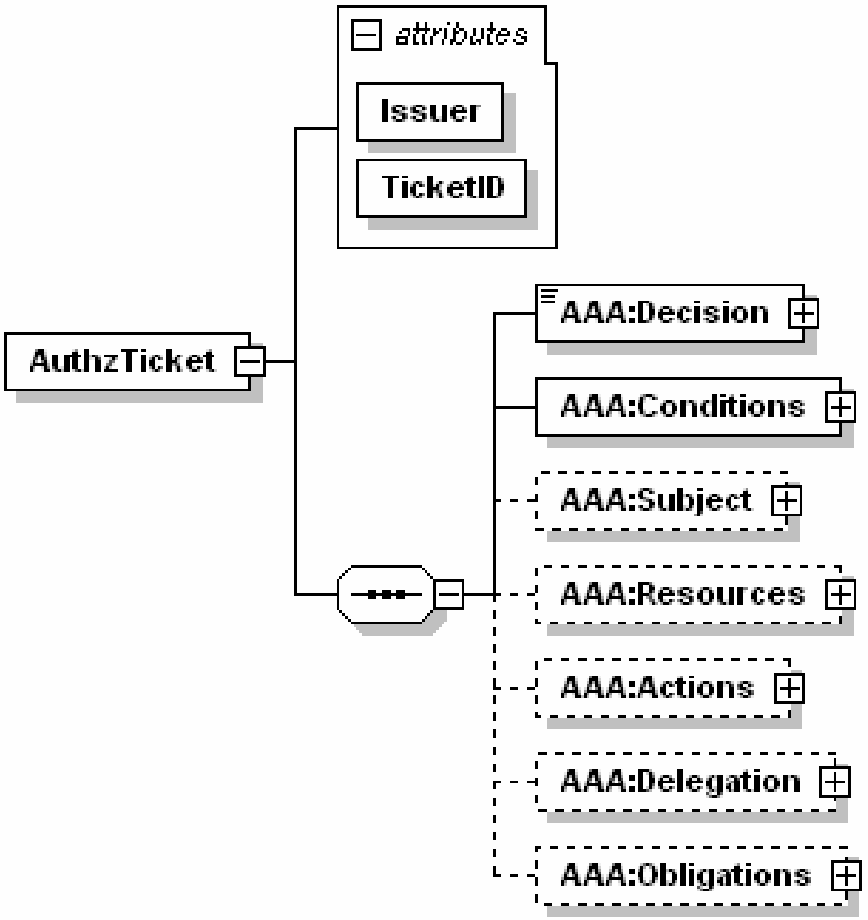


The proposed model intends to comply with both the generic AAA-AuthZ framework and XACML AuthZ model

- ContextHandler functionality can be extended to support all communications between PEP-PDP and with other modules

TTVS – Ticket and token validation and handling service

AuthZ ticket/assertion for extended security context management – Data model (1) - Top elements



Required functionality to support multidomain provisioning scenarios

- Allows easy mapping to SAML and XACML related elements

Allows multiple Attributes format (semantics, namespaces)

Establish and maintain Trust relations between domains

- Including Delegation

Ensure Integrity of the AuthZ decision

- Keeps AuthN/AuthZ context
- Allow Obligated Decisions (e.g. XACML)

Confidentiality

- Creates a basis for user-controlled Secure session

AuthZ ticket main elements

<Decision> element - holds the PDP AuthZ decision bound to the requested resource or service expressed as the ResourceID attribute.

<Conditions> element - specifies the validity constrains for the ticket, including validity time and AuthZ session identification and additionally context

<ConditionAuthzSession> (extendable) - holds AuthZ session context

<Subject> complex element - contains all information related to the authenticated Subject who obtained permission to do the actions

<Role> - holds subject's capbilities

<SubjectConfirmationData> - typically holds AuthN context

<SubjectContext> (extendable) - provides additional security or session related information, e.g. Subject's VO, project, or federation.

<Resources>/<Resource> - contains resources list, access to which is granted by the ticket

<Actions>/<Action> complex element - contains actions which are permitted for the Subject or its delegates

<Delegation> element – defines who the permission and/or capability are delegated to: another DelegationSubjects or DelegationCommunity

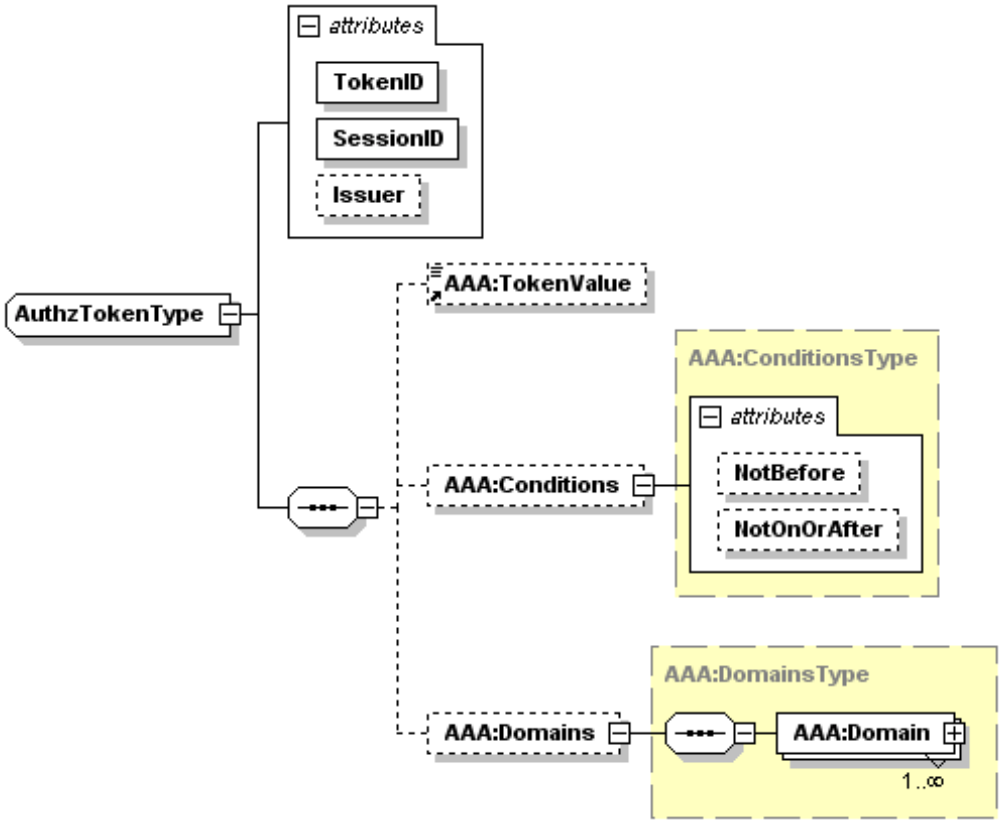
- attributes define restriction on type and depth of delegation

<Obligations>/<Obligation> element - holds obligations that PEP/Resource should perform in conjunction with the current PDP decision.

AuthZ ticket format (proprietary) for extended security context management

```
<AAA:AuthzTicket xmlns:AAA="http://www.aaauthreach.org/ns/#AAA" Issuer="urn:cnl:trust:tickauth:pep"
    TicketID="cba06d1a9df148cf4200ef8f3e4fd2b3">
  <AAA:Decision ResourceID="http://resources.collaboratory.nl/Philips_XPS1">Permit</AAA:Decision>
    <!-- SAML mapping: <AuthorizationDecisionStatement Decision="*" Resource="*"> -->
  <AAA:Actions>
    <AAA:Action>cnl:actions:CtrlInstr</AAA:Action>     <!-- SAML mapping: <Action> -->
    <AAA:Action>cnl:actions:CtrlExper</AAA:Action>
  </AAA:Actions>
  <AAA:Subject Id="subject">
    <AAA:SubjectID>WHO740@users.collaboratory.nl</AAA:SubjectID>    <!-- SAML mapping: <Subject>/<NameIdentifier> -->
    <AAA:SubjectConfirmationData>IGhA11vwa8YQomTgB9Ege9JRNnld84AggaDkOb5WW4U=</AAA:SubjectConfirmationData>
     <!-- SAML mapping:  EXTENDED <SubjectConfirmationData/> -->
    <AAA:Role>analyst</AAA:Role>
     <!-- SAML mapping:  <Evidence>/<Assertion>/<AttributeStatement>/<Assertion>/<Attribute>/<AttributeValue> -->
    <AAA:SubjectContext>CNL2-XPS1-2005-02-02</AAA:SubjectContext>
     <!-- SAML mapping: <Evidence>/<Assertion>/<AttributeStatement>/<Assertion>/<Attribute>/<AttributeValue> -->
  </AAA:Subject>
  <AAA:Delegation MaxDelegationDepth="3" restriction="subjects">
    <!-- SAML mapping:  LIMITED <AudienceRestrictionCondition> (SAML1.1), or <ProxyRestriction>/<Audience> (SAML2.0)  -->
    <AAA:DelegationSubjects> <AAA:SubjectID>team-member-2</AAA:SubjectID> </AAA:DelegationSubjects>
  </AAA:Delegation>
  <AAA:Conditions NotBefore="2006-06-08T12:59:29.912Z" NotOnOrAfter="2006-06-09T12:59:29.912Z" renewal="no">
    <!-- SAML mapping: <Conditions NotBefore="*" NotOnOrAfter="*"> -->
    <AAA:ConditionAuthzSession PolicyRef="PolicyRef-GAAA-RBAC-test001" SessionID="JobXPS1-2006-001">
     <!-- SAML mapping: EXTENDED <SAMLConditionAuthzSession PolicyRef="*" SessionID="*"> -->
      <AAA:SessionData>put-session-data-Ctx-here</AAA:SessionData>  <!-- SAML EXTENDED: <SessionData/> -->
    </AAA:ConditionAuthzSession>
  </AAA:Conditions>
  <AAA:Obligations>
    <AAA:Obligation>put-policy-obligation(2)-here</AAA:Obligation>  <!-- SAML EXTENDED: <Advice>/<PolicyObligation> -->
    <AAA:Obligation>put-policy-obligation(1)-here</AAA:Obligation>
  </AAA:Obligations>
</AAA:AuthzTicket>
<ds:Signature> <ds:SignedInfo/> <ds:SignatureValue>e4E27kNwEXoVdnXIBpGVjpaBGVY71Nypos...</ds:SignatureValue></ds:Signature>
```

General XML Token Format – Access and Pilot Tokens



- Required functionality to support multidomain provisioning scenarios
  - Allows easy mapping to SAML and XACML related elements
- Allows multiple Attributes format (semantics, namespaces)
- Establish and maintain Trust relations between domains
  - Including Delegation
- Ensure Integrity of the AuthZ decision
  - Keeps AuthN/AuthZ context
  - Allow Obligated Decisions (e.g. XACML)
- Confidentiality
  - Creates a basis for user-controlled Secure session

Pilot Token Types

Type 0 – Access token (refers to the reserved resources context)

Type 1 – Container for communicating the GRI during the reservation stage

- Contains the mandatory SessionId=GRI attribute and an optional Condition element

Type 2 – Origin/requestor authenticating token

- TokenValue element contains a value that can be used as the authentication value for the token origin
- TokenValue may be calculated of the GRI (optionally concatenated with the Issuer's domainId) by applying e.g. HMAC function with the requestor's symmetric or private key.

Type 3 – Extends Type 2 with the Domains element that allows collecting domains security context information when passing multiple domains during the reservation process

- Domains' information may include the previous token and the domain's trust anchor or public key.

Type 4 – Used at the deployment stage and can communicate between domains security context information about all participating in the provisioned lightpath or network infrastructure resources

- Can be used for programming/setting up a TVS infrastructure for consistent access control tokens processing at the resource access stage

Access XML token format - Example

```
<AAA:AuthzToken xmlns:AAA="http://www.aaauthreach.org/ns/#AAA"
   Issuer="urn:aaa:gaaapi:token:TVS"
   SessionId="a9bcf23e70dc0a0cd992bd24e37404c9e1709afb"
   TokenId="d1384ab54bd464d95549ee65cb172eb7">
<AAA:TokenValue>ebd93120d4337bc3b959b2053e25ca5271a1c17e</AAA:TokenValue>
    <AAA:Conditions NotBefore="2007-08-12T16:00:29.593Z" NotOnOrAfter="2007-
   08-13T16:00:29.593Z"/>
</AAA:AuthzToken>
```

where

SessionId = GRI (Global Reservation Id)

TokenId – unique identifier (serving for logging and accountability)

TokenValue – generated securely from GRI or AuthzTicket (digital SignatureValue)

- The element <TokenValue> and attributes SessionId and TokenId are mandatory, and the element <Conditions> and attributes Issuer, NotBefore, NotOnOrAfter are optional
- Binary token contains just two values – TokenValue and GRI

TVS functionality – Access control and signalling

- Basic TVS functionality is checking validity of an access token received from the PEP or AuthZ gateway/service
  - Extended TVS functionality allow token re-building when requesting service from the next domain
  - Additionally, TVS may be required to support token or token key distribution at the reservation stage or at the stage of the reserved resource deployment
- TVS supports pilot tokens handling functionality used during the reservation stage
  - Can be used for building dynamic security association of the reserved resources
- Token building (TB) function generates the token as derivative from the GRI and token key (which can also be generated based on GRI)
  - Additionally, TB should allow generating token dynamically using token key and variable dataflow data, e.g. IP packets payload as in case of TBS-IP

TVS Implementation (using shared secret)

TVS is implemented as a component of the GAAA Toolkit Java library

- Supports token based AuthZ enforcement mechanism and signalling
- TVS related classes are organised as a **org.aaaarch.gaaapi.tvs** package. All interfaces are supported by corresponding method of the TVS.java class
- Can be integrated into the target network provisioning systems and applications, in particular OSCARS and DRAGON
- Important practical convention
  - GRI is generated in the first domain or by the Reservation service

The token generation and handling model is based on the shared secret HMAC-SHA1 algorithm:

`TokenKey = HMAC(GRI, tb_secret)`

where GRI – global reservation identifier,
tb_secret – shared Token Builder secret.

A token is created in a similar way but using TokenKey as a HMAC secret:

`TokenValue = HMAC(GRI, TokenKey)`

This algorithm allows for chaining token generation and validation process

`GRI-TokenKey-TokenValue => LRI-l_TokenKey-l_Token`

XACML-NRP Profile

XACML Authorisation Interoperability profile for Network Resource Provisioning

- Defines a number of Subject, Resource, Action, Environment attributes used in the XACML policy definition
- Defines policy obligations format and handling model
- Recent update (July 2008) - http://staff.science.uva.nl/~demch/projects/aaauthreach/draft-interop-xacml-nrp-profile-012.pdf
- Also a part of the Phosphorus project D4.3.1 deliverable
- Reference implementation in the GAAA-TK library
- Considered as an extension of the XACML-Grid profile
  - "An XACML Attribute and Obligation Profile for Authorization Interoperability in Grids" (Joint project by EGEE, OSG, GT). Version 1.0, May 16, 2008 - https://edms.cern.ch/document/929867/1

XACML Policy format

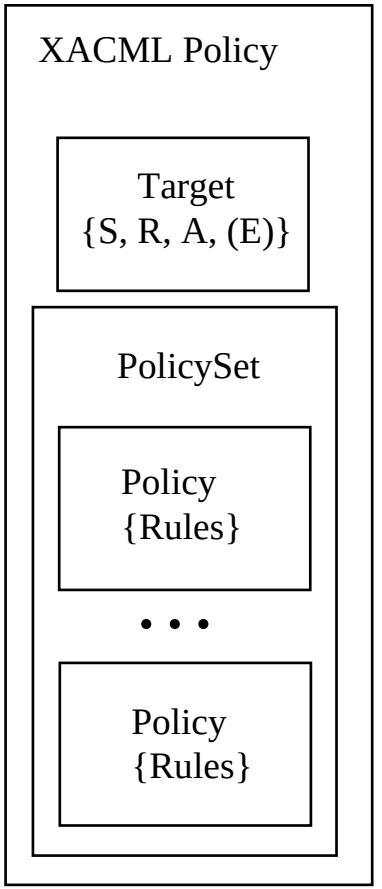
XACML standard specifies XACML policy format and XACML request/response messages

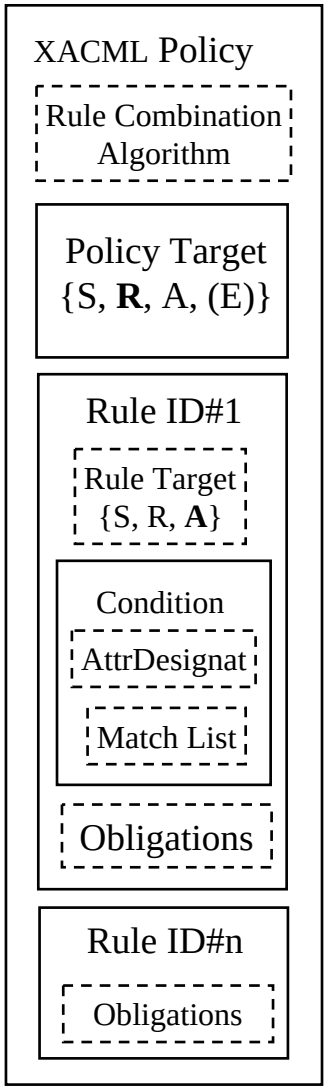
Policy consists of Policy Target and Rules

- Policy Target is defined for the tuple Subject-Resource-Action (-Environment)
- Policy Rule consists of Conditions and may contain Obligations
- Obligation defines actions to be taken by PEP on Policy decision by PDP

Policy obligation use examples

- Account mapping
- Quota or credit assignment
- Logging, accounting





XACML Request message - Example

```
<xacml-context:Request xmlns:xacml="urn:oasis:names:tc:xacml:1.0:policy" xmlns:xacml-
    context="urn:oasis:names:tc:xacml:1.0:context" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance" xsi:schemaLocation="urn:oasis:names:tc:xacml:1.0:context aaa-msg-xacml-01.xsd">
  <xacml-context:Subject Id="subject"    SubjectCategory="urn:oasis:names:tc:xacml:1.0:subject-
    category:access-subject">
     <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
     DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
       <xacml-context:AttributeValue>WHO740@users.project.organisation.nl</xacml-
     context:AttributeValue> </xacml-context:Attribute>

     <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-confdata"
     DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
       <xacml-context:AttributeValue>2SeDFGVHYTY83ZXxEdsweOP8Iok)yGHxVfHom90</xacml-
     context:AttributeValue> </xacml-context:Attribute>

     <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-role"
     DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
       <xacml-context:AttributeValue>Analyst</xacml-context:AttributeValue>
     </xacml-context:Attribute> </xacml-context:Subject>

  <xacml-context:Resource>
     <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
     DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="admin@gaaa.virtlab.nl">
       <xacml-context:AttributeValue>Resource-ID-here</xacml-context:AttributeValue>
     </xacml-context:Attribute>   </xacml-context:Resource>

     <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
     DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="admin@gaaa.collaboratory.nl">
       <xacml-context:AttributeValue>assign-time</xacml-context:AttributeValue>
     </xacml-context:Attribute>
  </xacml-context:Action> </xacml-context:Request>
```

Subject related Attributes

Attribute name	**Attribute ID**	**Full XACML attributeId semantics (ns-prefix = http://authz-interop.org/nrp/xacml)**
Subject ID	subject-id	{ns-prefix} /subject/subject-id http://authz-interop.org/nrp/xacml/subject/subject-id
Subject confirmation	subject-confdata	http://authz-interop.org/nrp/subject/subject-confdata
Subject context	subject-context	http://authz-interop.org/nrp/subject/subject-context
Subject group	subject-group	http://authz-interop.org/nrp/subject/subject-group
Subject role	subject-role	http://authz-interop.org/nrp/subject/subject-role
Subject federation	federation	http://authz-interop.org/nrp/subject/federation

Resource related attributes

Attribute name	Attribute ID	Full XACML attributeId semantics (ns-prefix = http://authz-interop.org/nrp/xacml)
Domain ID	domain-id	{ns-prefix} /resource/domain-id
Subdomain	subdomain	{ns-prefix} /resource/sub-domain
VLAN	vlan	{ns-prefix} /resource/vlan
TNA	tna (+ tna-prefix)	{ns-prefix} /resource/tna-prefix/tna
Node	node	{ns-prefix} /resource/node
Link	link-id	{ns-prefix} /resource/link-id
avrDelay	delay	{ns-prefix} /resource/delay
maxBW	bandwidth-max	{ns-prefix} /resource/bandwidth
Resource type	resource-type	{ns-prefix} /resource/resource-type ({ns-prefix} /resource/device)
Resource federation	federation	{ns-prefix} /resource/federation

- Domain ID (network domain)
- Subdomain (or relationship)
- VLAN
- Node or TNA and TNA prefix, or
- Interface ID
- Device or resource-type
- Link ID
- Link parameters: average delay and maximum bandwidth
- ReservationEPR that may directly or indirectly define the resource federation or security/ administrative domain
- Federation that defines a number of domains or nodes sharing common policy and attributes

XACML Obligations - Definition

Policy Obligation is one of the policy enforcement mechanisms

- ***Obligations*** are a set of operations that must be performed by the ***PEP*** in conjunction with an ***authorization decision*** [XACML2.0]

Obligations semantics is not defined in the XACML policy language but left to bilateral agreement between a PAP and the PEP

PEPs that conform with XACMLv2.0 are required to deny access unless they understand and can discharge all of the <Obligations> elements associated with the applicable policy

Element <Obligations> / <Obligation>

- The <Obligation> element SHALL contain an ***identifier*** (in the form of URI) for the obligation and a set of attributes that form arguments of the action defined by the obligation. The FulfillOn attribute SHALL indicate the effect for which this obligation must be fulfilled by the PEP

XACML Obligations – Implementation suggestions

Obligation = Apply (TargetAttribute, Operation (Variables)), or
Obligation = Apply (TargetAttribute, Operation (Variables), Chronicle)

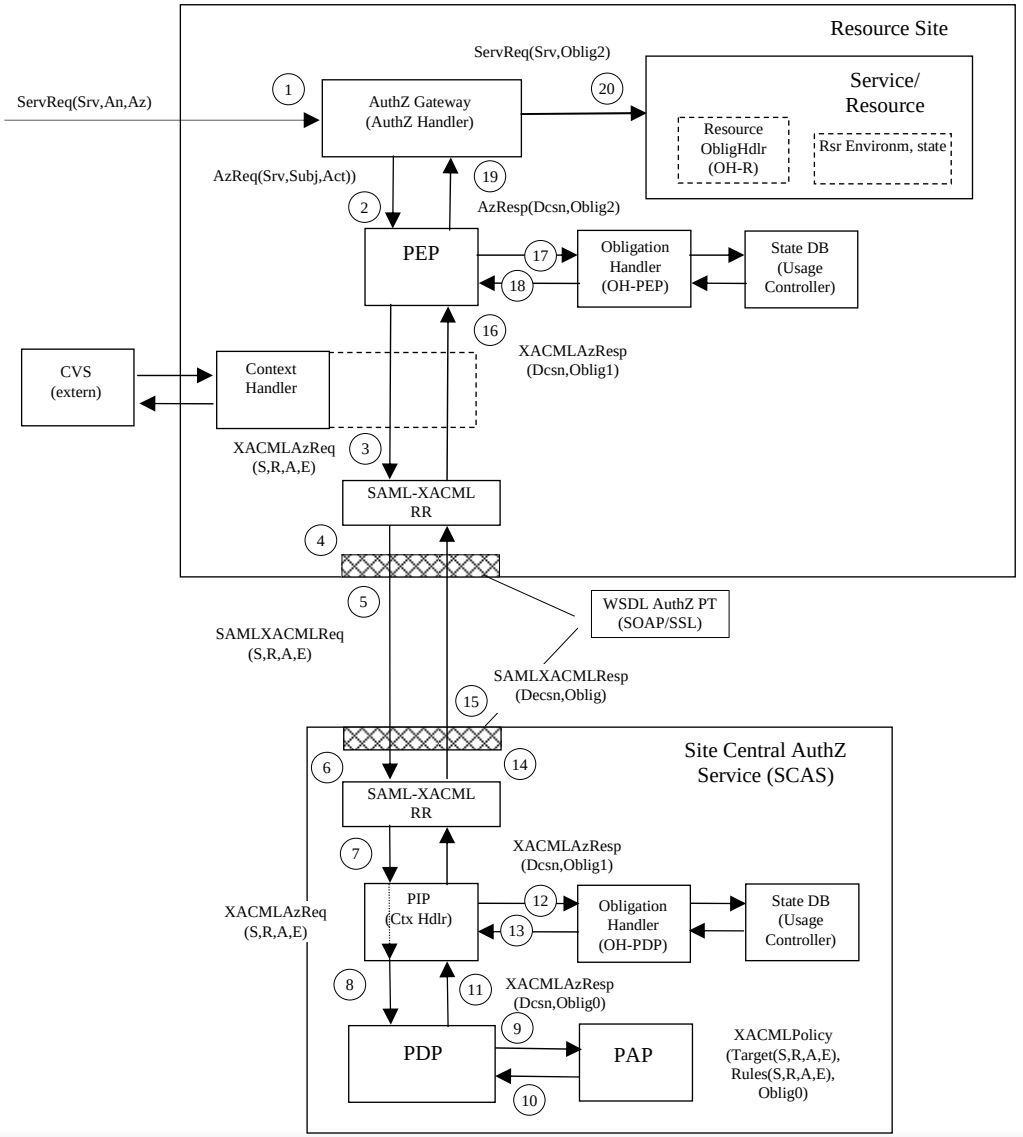
Obligations enforcement scenarios

- Obligations are enforced by PEP at the time of receiving obligated AuthZ decision from PDP
- Obligations are enforced at later time when the requestor accesses the resource or service
- Obligations are enforced before or after the resource or service accessed/delivered/consumed

Obligation handling model proposed in the process of interoperability workshop between OSG, EGEE, and Globus

- ObligationId (of type URI) has to be mapped to a specific handler that is called by the PEP
- Obligation parameter values are passed to handler
- Handler returns True/False that determines PEP's Permit/Deny

Proposed Obligations Handling Reference Model



Generic AuthZ service model

PEP – Policy Enforcement Point

PDP – Policy Decision Point

PAP – Policy Authority Point

OH – Obligation Handler

CtxHandler – Context Handler

(S, R, A, E) – components of the AuthZ request (Subject, Resource, Action, Environment)

Obligations Handling Stages

Obligation0 = tObligation => Obligation1 (“OK?”, (Attributes1 v Environments1))
=> Obligation2 (“OK?”, (Attributes2 v Environments2))
=> Obligation3 (Attributes3 v Environments3)

Obligation0 – (stateless or template)

Obligations are returned by the PDP in a form as they are written in the policy. These obligations can be also considered as a kind of templates or instructions, tObligation.

Obligation1 and Obligation 2

Obligations have been handled by Obligation handler at the SCAS/PDP side or at the PEP side, depending on implementation. Templates or instructions of the Obligation0 are replaced with the real attributes in Obligation1/2, e.g. in a form of “name-value” pair.

- The result of Obligations processing/enforcement is returned in a form of modified AuthzResponce (Obligation1) or global Resource environment changes
- Obligation handler should return notification about fulfilled obligated actions, e.g. in a form of Boolean value “False” or “True”, which will be taken into account by PEP or other processing module to finally permit or deny service request by PEP.
- Note. Obligation1 handling at the SCAS or PDP side allows stateful PDP/SCAS.

Obligation3

Final stage when an Obligation actually takes effect (Obligations “termination”). This is done by the Resource itself or by services managed/controlled by the Resource.

XACML Obligations – Examples of expression for pool account mapping in Grid – Option 1 (simple, used in XACML-Grid)

```
<!-- Obligations format option 1 (simple): UID, GID explicitly mentioned as
   separate XML elements inside AttributeAssignment element -->
<xacml:Obligations>
   <xacml:Obligation
        ObligationId=http://authz-interop.org/xacml/obligation/uidgid
        FulfillOn="Permit">
     <xacml:AttributeAssignment
          AttributeId=http://authz-interop.org/xacml/attribute/posix-uid
          DataType="http://www.w3.org/2001/XMLSchema#integer">
       2501</xacml:AttributeAssignment>
     <xacml:AttributeAssignment
          AttributeId=http://authz-interop.org/xacml/attribute/posix-gid
          DataType="http://www.w3.org/2001/XMLSchema#integer">
       2101</xacml:AttributeAssignment>
   </xacml:Obligation>
<xacml:Obligations>
```

XACML Obligations – Examples of expression for pool account mapping in Grid – Option 2

```
<Obligations>
<Obligation ObligationId="http://authz-interop.org/xacml/obligation/map.poolaccount"
   FulfillOn="Permit">
<!-- Specifies to what kind of attribute the next ‘map.to’ action is applied to -->
<AttributeAssignment
AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute: requesting-subject"
DataType="http://www.w3.org/2001/XMLSchema#string">
    &lt;SubjectAttributeDesignator
    AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
    DataType="http://www.w3.org/2001/XMLSchema#string"/&gt;
</AttributeAssignment>

<!-- This is actual account attribute name/value to which it should be mapped -->
<AttributeAssignment
    AttributeId="http://authz-interop.org/xacml/obligation/attribute/uidgid"
    DataType="http://www.w3.org/2001/XMLSchema#string">
  &lt;UnixId DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
    okoeroo&gt;UnixId&gt;
  &lt; GroupPrimary DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
    computergroup&gt;GroupPrimary&gt;
  &lt;GroupSecondary DataType="http://www.w3.org/2001/XMLSchema#string"&gt;
    datagroup&gt;GroupSecondary&gt;
</AttributeAssignment>
</Obligation>
</Obligations>
```

Implementation – GAAA-TK pluggable Java library

- All proposed AuthZ mechanisms and service components are implemented as the GAAA Toolkit pluggable Java library (GAAA-TK)
  - Uses best of the gLite (and Globus Toolkit) AuthZ service implementation
  - Supports XACML policy profile and SAML-XACML profile AuthZ service protocol
  - Part of the Phosphorus project deliverable D.4.3.1 - "GAAA toolkit pluggable components and XACML policy profile for ONRP“ http://staff.science.uva.nl/~demch/worksinprogress/Phosphorus-WP4-D4.3.1-GAAA-TK-library-NRP-v02.pdf
- Integrated into the Phosphorus project Network Service Plane (NSP) test-bed and uses simple XACML policy model
- Currently being integrated into Grid-enabled GMPLS Control Plane service (Phosphorus project) that motivates a number of the specific cross-domain use case scenarios

Future developments

- Defining network topology aware XACML-NRP policy model
- Combining network and Grid resources into one provisioning and AuthZ workflow/session
- Extend AuthZ session management model and related AuthZ ticket functionality and XACML policy model to support multidomain reservation process and (restricted) delegation
- GAAA-TK library interoperability and integration with major Grid middleware and GN2/NRENs eduGAIN AAI
- Using Identity Based Cryptography (IBC) for cross-domain trust relations management

PKI vs Identity Based Cryptography (IBC)

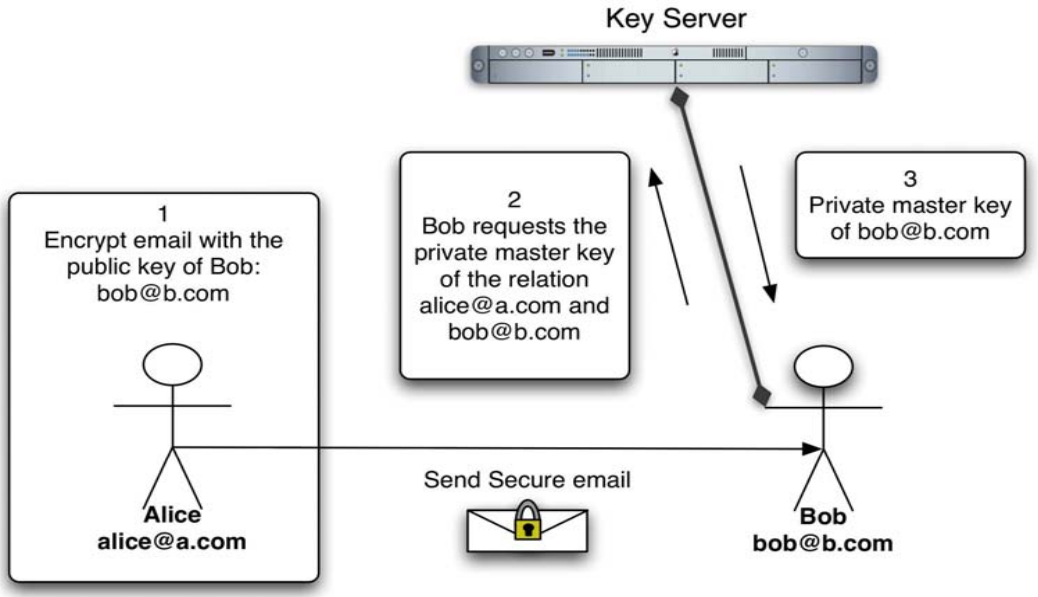
Uses publicly known remote entity's identity as a public key to send encrypted message or initiate security session

- Idea was proposed by Shamir in 1984 as an alternative to PKI and implementation by Dan Boneh and Matthew K. Franklin in 2001
- Identity can be email address, domain name, IP address
- Allows conditional private key generation

Requires infrastructure different from PKI but domain based (doesn't require trusted 3rd party outside of domain)

- Parties may encrypt messages (or verify signatures) with no prior distribution of keys between individual participants
- Private key generation service (KGS)
  - Generates private key to registered/authenticated users/entities
  - To operate, the KGS first publishes a master public key, and retains the corresponding **master private key** (referred to as *master key*).
  - Given the master public key, any party can compute a public key corresponding to the identity *ID* by combining the master public key with the identity value.
- Exchange inter-domain trust management problem to intra-domain trust

Identity Based Cryptography (IBC) - Operation



Four algorithms form a complete IBE system (as proposed by Dan Boneh and Matthew K. Franklin):

Setup: This algorithm is run by the PKG one time for creating the whole IBE environment.

- The master key is kept secret and used to derive users' private keys, while the system parameters are made public. It accepts a security parameter k (i.e. binary length of key material) and outputs:
- A set P of system parameters, including the message space and ciphertext space M and C, a master key Km (master) .

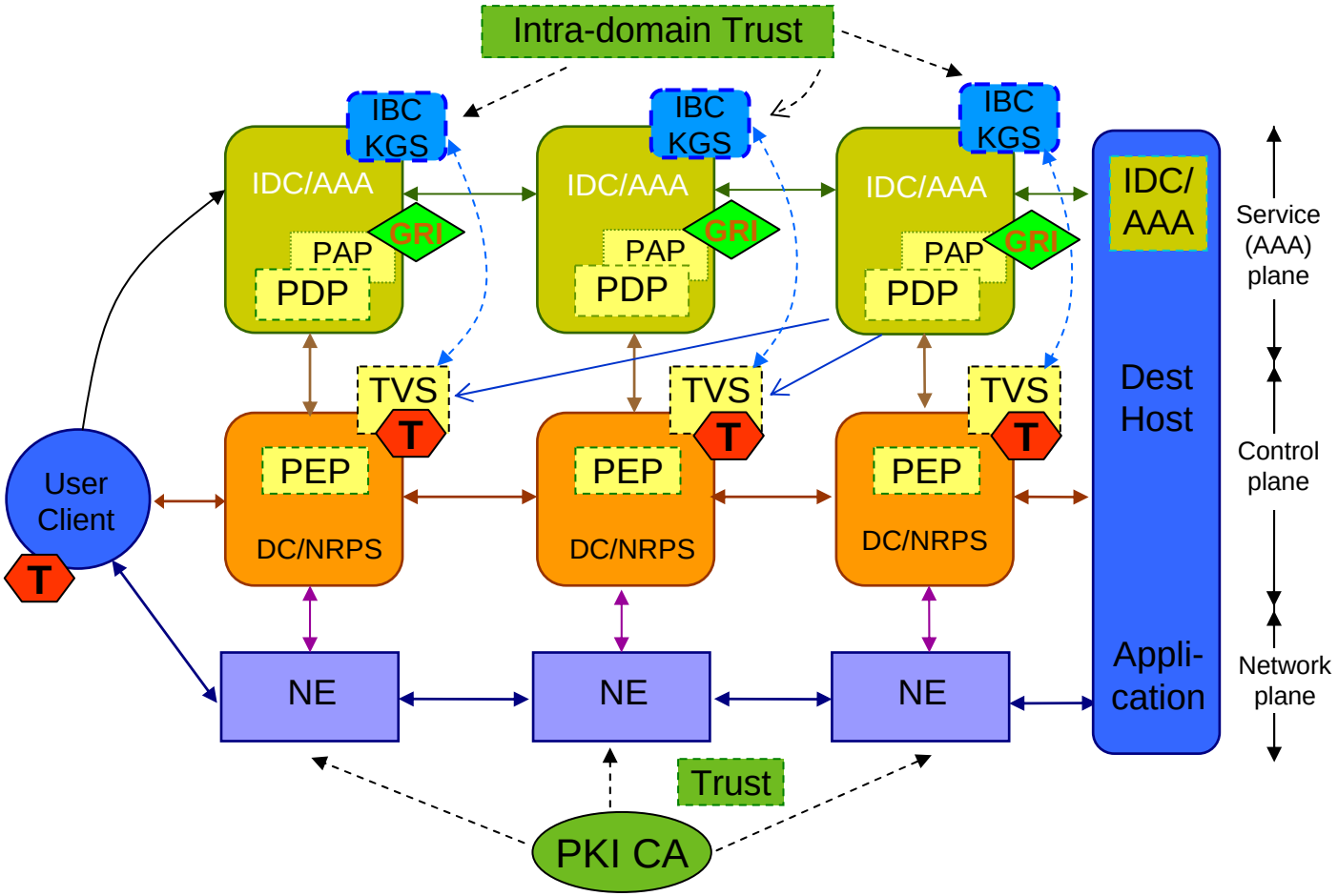
Extract: This algorithm is run by the PKG when a user requests his private key.

- It takes as input P, Km and an identifier ID={0,1} and returns the private key D for user ID.
- Requires strong authentication and out of IBE model scope

Encrypt: Takes P, a message m={M} and ID={0,1} and outputs the encryption c={C}.

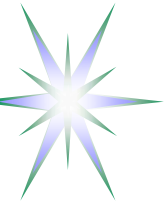
Decrypt: Accepts d, P and c={C} and returns m={M}

Identity Based Cryptography (IBC) infrastructure operation when distributing token keys in multidomain NRP



- Uses intra-domain trust relation without prior public key exchange
- Simplifies key management problem
- Allows flexibility in deploying/configuring intra-domain network path/infrastructure
- Used at deployment stage

- IBC KGS are setup independently but publish their public parameters

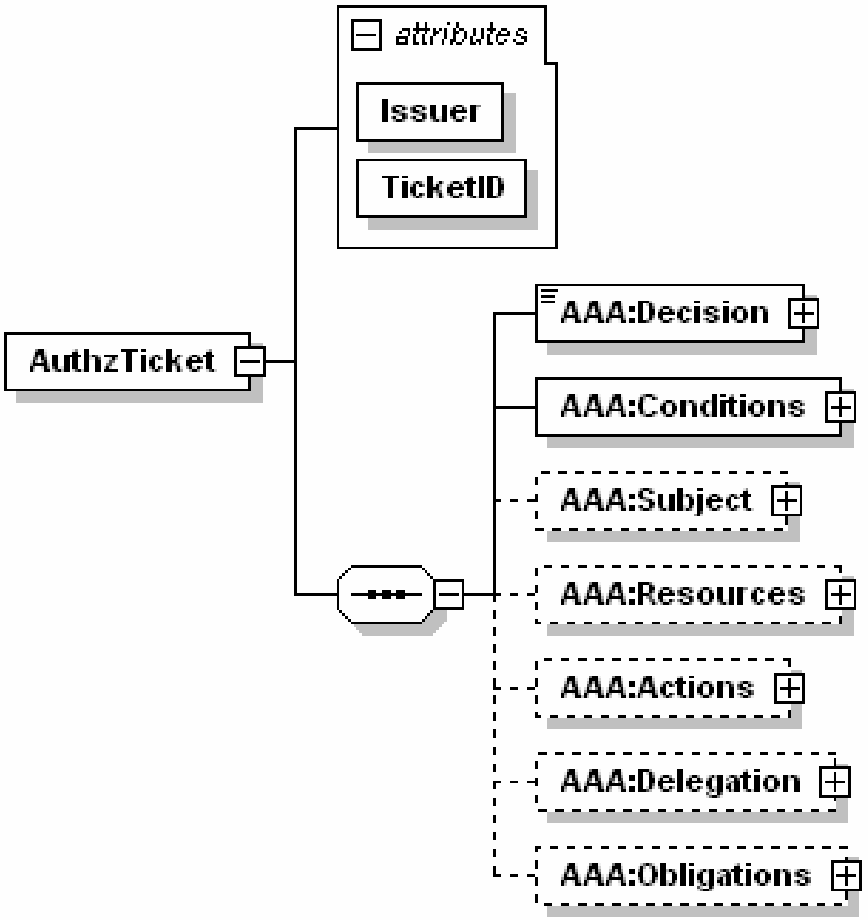


Questions and Discussion

Additional information

- Detailed AuthZ ticket data model
- SAML-XACML Request-Response format
- XACML-NRP profile

AuthZ ticket/assertion for extended security context management – Data model (1) - Top elements



Required functionality to support multidomain provisioning scenarios

- Allows easy mapping to SAML and XACML related elements

Allows multiple Attributes format (semantics, namespaces)

Establish and maintain Trust relations between domains

- Including Delegation

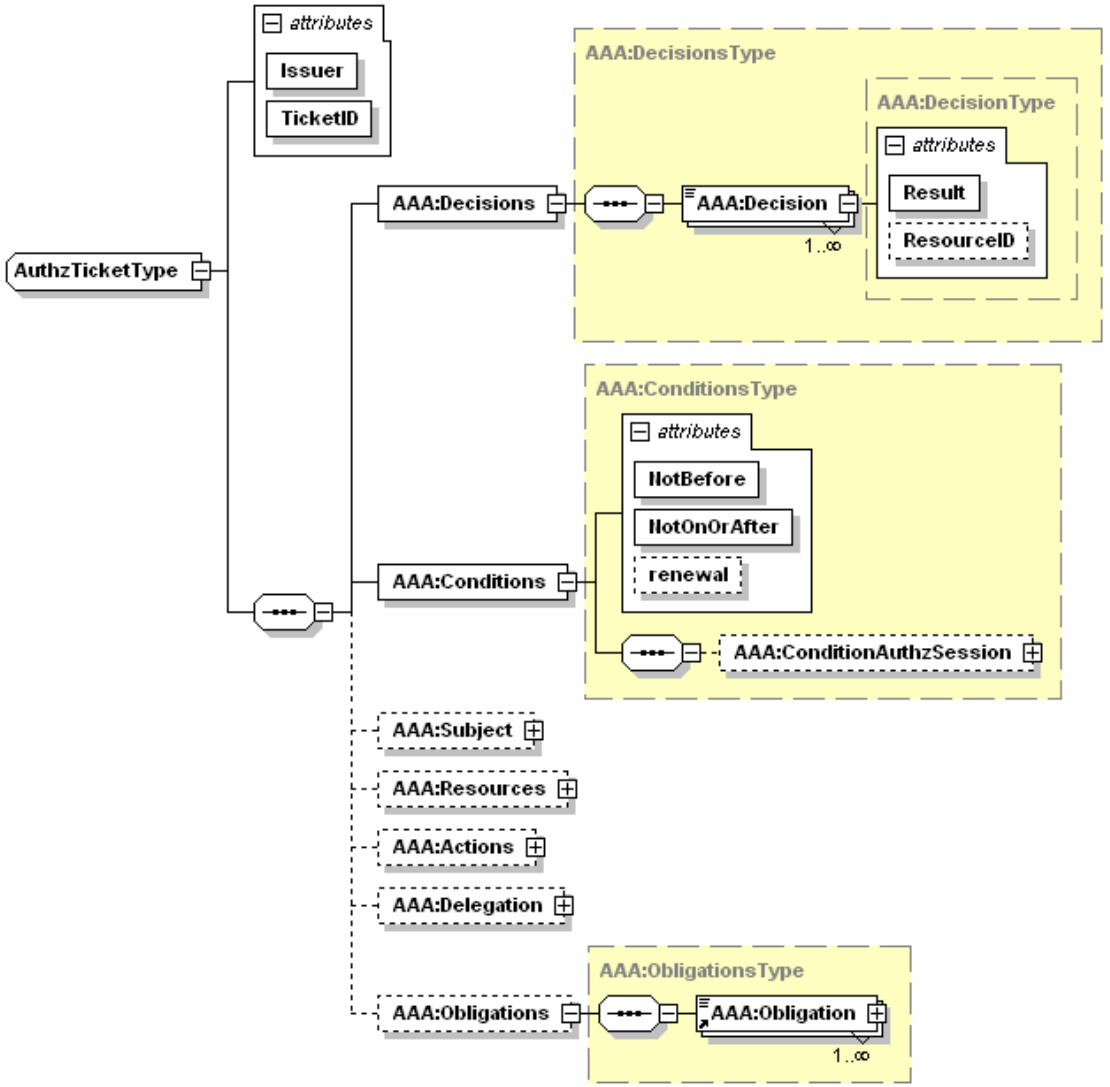
Ensure Integrity of the AuthZ decision

- Keeps AuthN/AuthZ context
- Allow Obligated Decisions (e.g. XACML)

Confidentiality

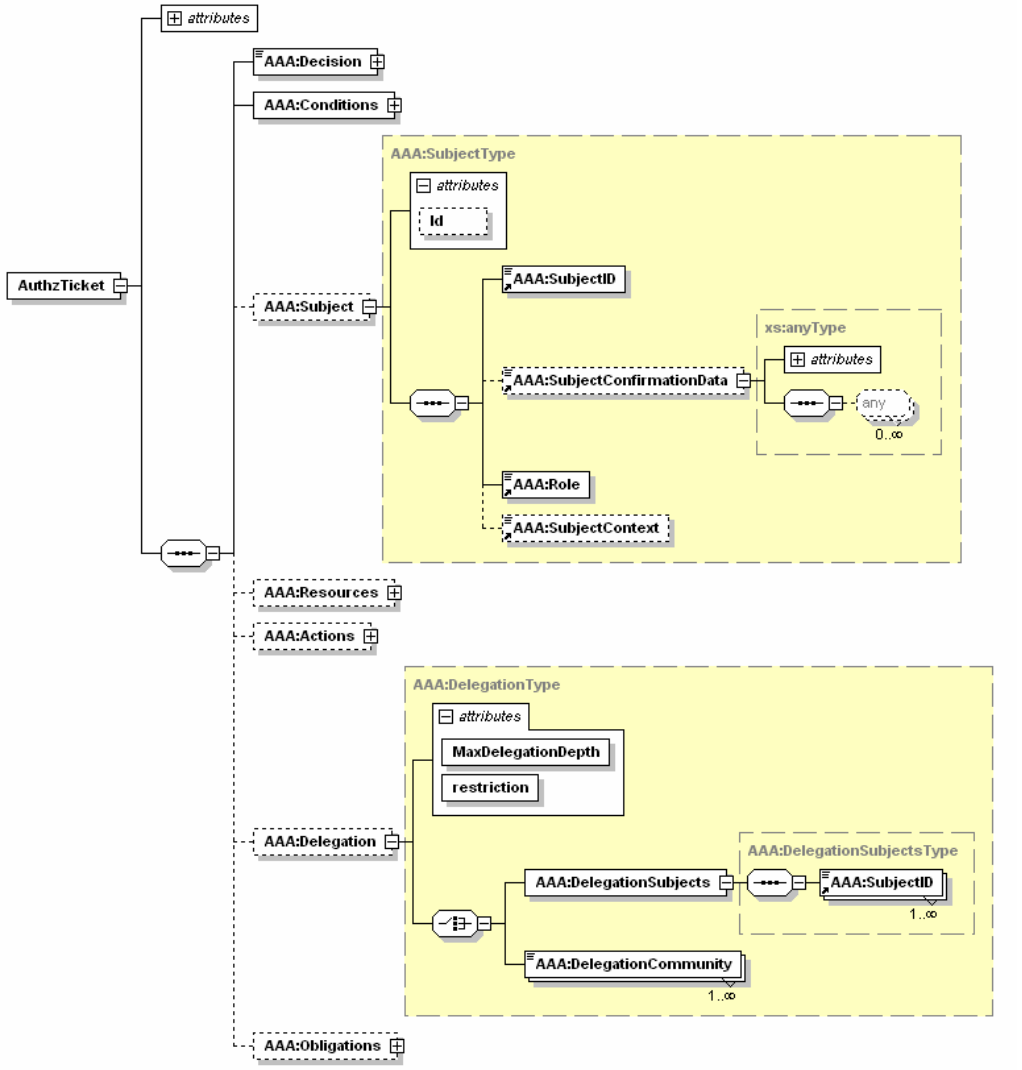
- Creates a basis for user-controlled Secure session

AuthZ ticket Data model (2) - Mandatory elements



- TicketID attribute
- Decisions element and ResourceID attribute
- Conditions Element and validity attributes
- Extensible element ConditionAuthzSession
  - Any AuthZ session related data

AuthZ ticket Data model (3) – Subject and Delegation elements



- Subject element to keep AuthN security context and Subject Attributes
- Delegation element to allow permissions/AuthZ decision delegation to other Subjects or groups/community

AuthZ ticket main elements

<Decision> element - holds the PDP AuthZ decision bound to the requested resource or service expressed as the ResourceID attribute.

<Conditions> element - specifies the validity constrains for the ticket, including validity time and AuthZ session identification and additionally context

<ConditionAuthzSession> (extendable) - holds AuthZ session context

<Subject> complex element - contains all information related to the authenticated Subject who obtained permission to do the actions

<Role> - holds subject's capbilities

<SubjectConfirmationData> - typically holds AuthN context

<SubjectContext> (extendable) - provides additional security or session related information, e.g. Subject's VO, project, or federation.

<Resources>/<Resource> - contains resources list, access to which is granted by the ticket

<Actions>/<Action> complex element - contains actions which are permitted for the Subject or its delegates

<Delegation> element – defines who the permission and/or capability are delegated to: another DelegationSubjects or DelegationCommunity

- attributes define restriction on type and depth of delegation

<Obligations>/<Obligation> element - holds obligations that PEP/Resource should perform in conjunction with the current PDP decision.

AuthZ ticket format (proprietary) for extended security context management

```
<AAA:AuthzTicket xmlns:AAA="http://www.aaauthreach.org/ns/#AAA" Issuer="urn:cnl:trust:tickauth:pep"
    TicketID="cba06d1a9df148cf4200ef8f3e4fd2b3">
  <AAA:Decision ResourceID="http://resources.collaboratory.nl/Philips_XPS1">Permit</AAA:Decision>
    <!-- SAML mapping: <AuthorizationDecisionStatement Decision="*" Resource="*"> -->
  <AAA:Actions>
    <AAA:Action>cnl:actions:CtrlInstr</AAA:Action>     <!-- SAML mapping: <Action> -->
    <AAA:Action>cnl:actions:CtrlExper</AAA:Action>
  </AAA:Actions>
  <AAA:Subject Id="subject">
    <AAA:SubjectID>WHO740@users.collaboratory.nl</AAA:SubjectID>    <!-- SAML mapping: <Subject>/<NameIdentifier> -->
    <AAA:SubjectConfirmationData>IGhA11vwa8YQomTgB9Ege9JRNnld84AggaDkOb5WW4U=</AAA:SubjectConfirmationData>
     <!-- SAML mapping:  EXTENDED <SubjectConfirmationData/> -->
    <AAA:Role>analyst</AAA:Role>
     <!-- SAML mapping:  <Evidence>/<Assertion>/<AttributeStatement>/<Assertion>/<Attribute>/<AttributeValue> -->
    <AAA:SubjectContext>CNL2-XPS1-2005-02-02</AAA:SubjectContext>
     <!-- SAML mapping: <Evidence>/<Assertion>/<AttributeStatement>/<Assertion>/<Attribute>/<AttributeValue> -->
  </AAA:Subject>
  <AAA:Delegation MaxDelegationDepth="3" restriction="subjects">
     <!-- SAML mapping:  LIMITED <AudienceRestrictionCondition> (SAML1.1), or <ProxyRestriction>/<Audience> (SAML2.0)  -->
    <AAA:DelegationSubjects> <AAA:SubjectID>team-member-2</AAA:SubjectID> </AAA:DelegationSubjects>
  </AAA:Delegation>
  <AAA:Conditions NotBefore="2006-06-08T12:59:29.912Z" NotOnOrAfter="2006-06-09T12:59:29.912Z" renewal="no">
     <!-- SAML mapping: <Conditions NotBefore="*" NotOnOrAfter="*"> -->
    <AAA:ConditionAuthzSession PolicyRef="PolicyRef-GAAA-RBAC-test001" SessionID="JobXPS1-2006-001">
     <!-- SAML mapping: EXTENDED <SAMLConditionAuthzSession PolicyRef="*" SessionID="*"> -->
      <AAA:SessionData>put-session-data-Ctx-here</AAA:SessionData>  <!-- SAML EXTENDED: <SessionData/> -->
    </AAA:ConditionAuthzSession>
  </AAA:Conditions>
  <AAA:Obligations>
    <AAA:Obligation>put-policy-obligation(2)-here</AAA:Obligation>  <!-- SAML EXTENDED: <Advice>/<PolicyObligation> -->
    <AAA:Obligation>put-policy-obligation(1)-here</AAA:Obligation>
  </AAA:Obligations>
</AAA:AuthzTicket>
<ds:Signature> <ds:SignedInfo/> <ds:SignatureValue>e4E27kNwEXoVdnXIBpGVjpaBGVY71Nypos...</ds:SignatureValue></ds:Signature>
```

XACML-NRP Profile – Work in Progress

XACML Authorisation Interoperability profile for Network Resource Provisioning. Phosphorus technical document

- Recent release (July 2008) - http://staff.science.uva.nl/~demch/projects/aaauthreach/draft-interop-xacml-nrp-profile-012.pdf
- Also a part of the Phosphorus project D4.3.1 deliverable

Related document

"An XACML Attribute and Obligation Profile for Authorization Interoperability in Grids" (Joint project by EGEE, OSG, GT). Version 1.0, May 16, 2008.

- https://edms.cern.ch/document/929867/1

Basic use cases for policy definition in NRP

Use case 1: "User A is only allowed to use user endpoints X, Y and Z", or

Use case 2: "User A is only allowed to use endpoints in domain N and M"

Policy definition assumptions

- Users and resources are described/identified by their unique ID's and may have also assigned attributes, e.g.
  - User attrs: user group, role, federation
  - Resource attrs: domain/subdomain, resource type, level of service
- Users and resources (domains and endpoints) may be organised/associated into administrative and/or security domains or federations
  - A user and a resource can be a member of one or multiple associations
- Different domains and endpoints participating in network connection (for which the authorisation is requested) may belong to different federations or security associations
- Only authenticated user may have access to protected resources
  - User authentication is confirmed by issuing AuthZ assertion by trusted AuthN service or creating user related security context environment of the started process
- User authentication may be resulted in the following:
  - service or process session initiation;
  - release of the user attributes or credentials;
- Depending on the user attributes (federations, groups, roles) the user can be assigned specific level of service
  - To access a network resources a user identity may need to be mapped to a specific (pool) account

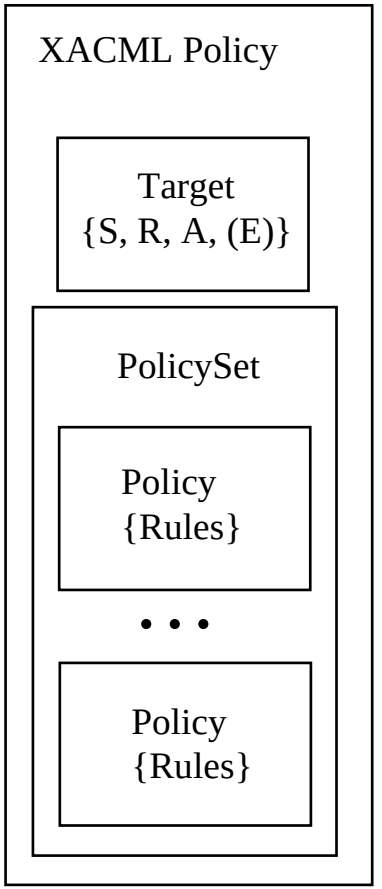
XACML Policy format

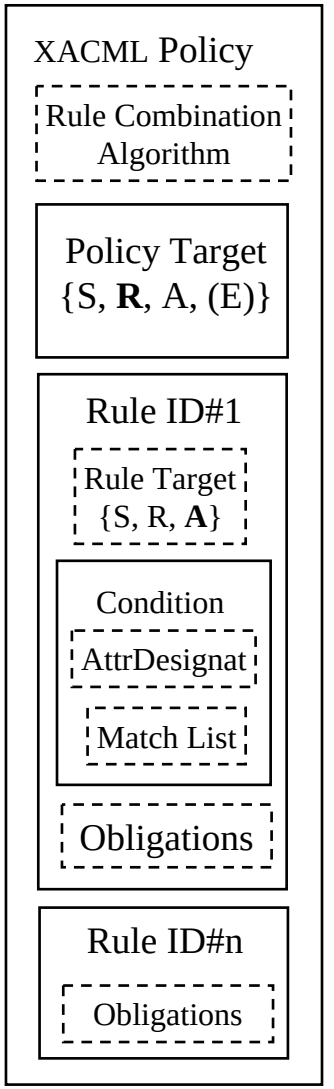
Policy consists of Policy Target and Rules

- Policy Target is defined for the tuple Subject-Resource-Action (-Environment)
- Policy Rule consists of Conditions and may contain Obligations
- Obligation defines actions to be taken by PEP on Policy decision by PDP

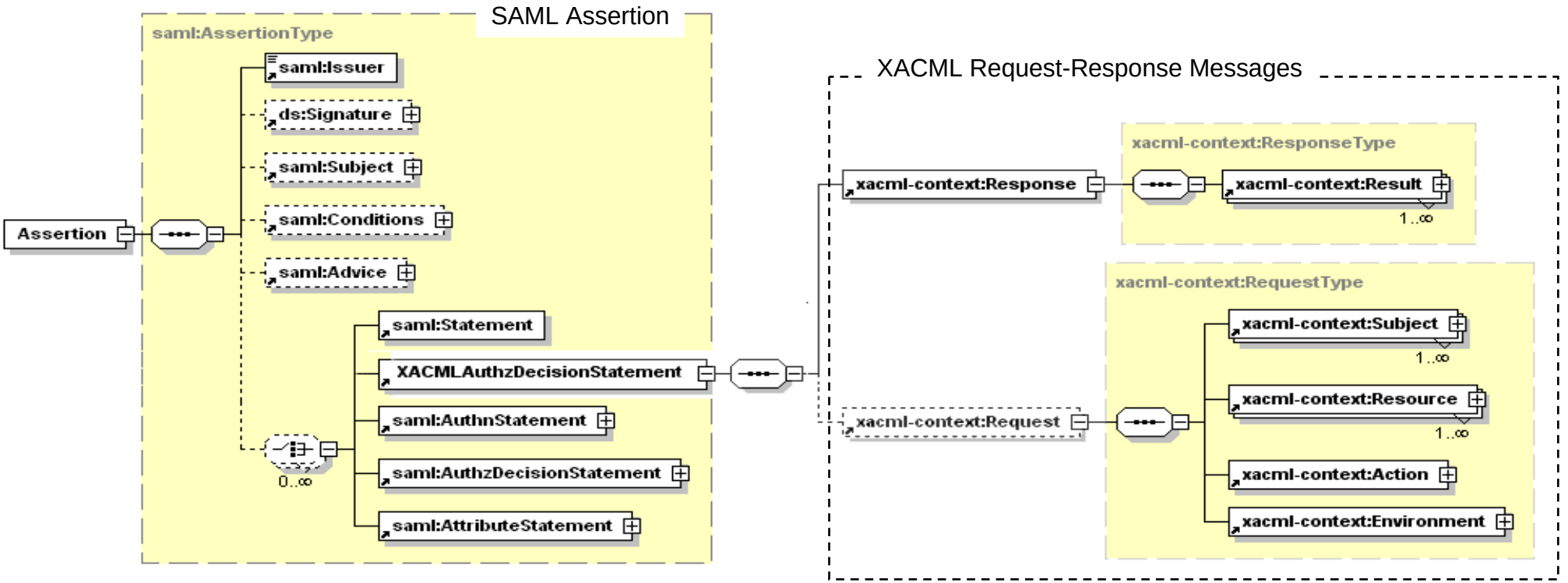
Policy obligation use examples

- Account mapping
- Quota or credit assignment
- Logging, accounting





SAML-XACML Request/Response messages



XACMLRequest (Resource, Subject, Action, Environment)

XACML Request-Response messages are enclosed into the SAML2.0 Assertion or SAML2.0 protocol messages

XACML Request message - Example

```
<xacml-context:Request xmlns:xacml="urn:oasis:names:tc:xacml:1.0:policy" xmlns:xacml-
    context="urn:oasis:names:tc:xacml:1.0:context" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:oasis:names:tc:xacml:1.0:context aaa-msg-xacml-01.xsd">
  <xacml-context:Subject Id="subject"    SubjectCategory="urn:oasis:names:tc:xacml:1.0:subject-category:access-
    subject">
    <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
    DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
      <xacml-context:AttributeValue>WHO740@users.project.organisation.nl</xacml-context:AttributeValue>
    </xacml-context:Attribute>
    <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subjconfdata"
    DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
      <xacml-context:AttributeValue>2SeDFGVHYTY83ZXxEdsweOP8Iok)yGHxVfHom90</xacml-context:AttributeValue>
    </xacml-context:Attribute>
    <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:role"
    DataType="http://www.w3.org/2001/XMLSchema#string" Issuer=" admin@gaaa.virtlab.nl ">
      <xacml-context:AttributeValue>Analyst</xacml-context:AttributeValue>
    </xacml-context:Attribute>
  </xacml-context:Subject>
  <xacml-context:Resource>
    <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
    DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="admin@gaaa.virtlab.nl">
      <xacml-context:AttributeValue>Resource-ID-here</xacml-context:AttributeValue>
    </xacml-context:Attribute>
  </xacml-context:Resource>
    <xacml-context:Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
    DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="admin@gaaa.collaboratory.nl">
      <xacml-context:AttributeValue>assign-time</xacml-context:AttributeValue>
    </xacml-context:Attribute>
  </xacml-context:Action>
</xacml-context:Request>
```

Resource related Attributes – Topology description formats

- Actually depends on the used/target topology description format
- 3 topology description formats were reviewed
  - Phosphorus NSP/WP1 topology description
  - NDL by UvA
  - OSCARS (currently used)

- Examples AuthZ decision request
  - Is user A allowed to access this reserved path given known (multidomain) network topology?
  - Needs to put some topology attributes into the policy definition

Example of the Resource attributes expression

Attribute name	**Attribute ID**	**Full XACML attributeId semantics (ns-prefix = http://authz-interop.org/nrp/xacml)**
Domain ID	domain-id	{ns-prefix} /resource/domain-id
Subdomain	subdomain	{ns-prefix} /resource/sub-domain
VLAN	vlan	{ns-prefix} /resource/vlan
TNA	tna (+ tna-prefix)	{ns-prefix} /resource/tna-prefix/tna
Node	node	{ns-prefix} /resource/node
Link	link-id	{ns-prefix} /resource/link-id
avrDelay	delay	{ns-prefix} /resource/delay
maxBW	bandwidth-max	{ns-prefix} /resource/bandwidth
Resource type	resource-type	{ns-prefix} /resource/resource-type ({ns-prefix} /resource/device)
Resource federation	federation	{ns-prefix} /resource/federation

- Domain ID (network domain)
- Subdomain (or relationship)
- VLAN
- Node or TNA and TNA prefix, or
- Interface ID
- Device or resource-type
- Link ID
- Link parameters: average delay and maximum bandwidth
- ReservationEPR that may directly or indirectly define the resource federation or security/ administrative domain
- Federation that defines a number of domains or nodes sharing common policy and attributes

Administrative vs Security domain vs Security Association

- Domains can be considered as administrative and security
  - Domains are more static
  - Administrative domain is managed by the resource owner (or user administration)
  - Security domain is defined by common trusted identity or attribute management authority
- Security association
  - Security association can be created dynamically, e.g. for managing project, resource provisioning agreement
    - VO or Shibboleth federation are two examples
  - Authorisation session

Subject related Attributes

Attribute name	**Attribute ID**	**Full XACML attributeId semantics (ns-prefix = http://authz-interop.org/nrp/xacml)**
Subject ID	subject-id	{ns-prefix} /subject/subject-id
Subject confirmation	subject-confdata	{ns-prefix} /subject/subject-confdata
Subject Context	subject-context	{ns-prefix} /subject/subject-context
Subject group	subject-group	{ns-prefix} /subject/subject-group
Subject role	subject-role	{ns-prefix} /subject/subject-role
Subject federation	Federation	{ns-prefix} /subject/federation

Action related Attributes and Enumerated values

Attribute name	**Attribute ID**	**Full XACML attributeId semantics (ns-prefix = http://authz-interop.org/nrp/xacml)**
Action ID	action-id	{ns-prefix} /action/action-id
Action type	action-type	{ns-prefix} /action/action-type/{value}

Attribute name	**Enumerated value**	**XACML attribute value (ns-prefix = http://authz-interop.org/nrp/xacml)**
Action type	create-path	{ns-prefix} /action/action-type/create-path
	activate-path	{ns-prefix} /action/action-type/activate-path
	cancel	{ns-prefix} /action/action-type/cancel
	access	{ns-prefix} /action/action-type/access

Environment related Attributes

- Last-domain conformation
- Authorisation context
  - AuthZ session credentials or AuthZ ticket
- Delegation or Obligations from the previous domain
  - User ID or group to which access is delegated
  - Actions which need to be taken when processing request or granting access

Policy Obligations

Suggested policy obligations for multidomain NRP

- User identity/account mapping
- Intra-domain VLAN mapping
- Accounting, logging
- Usability