Using SAML and XACML for Complex Authorisation Scenarios in Dynamic Resource Provisioning

Yuri Demchenko <demch@science.uva.nl>
System and Network Engineering Group
University of Amsterdam

ARES 2007 Conference
10-13 April 2007, Vienna

Outline

- Background: Target projects and AuthZ service basics
- General Complex Resource Provisioning (CRP) model
- GAAA-AuthZ components to support dynamic security context management
- AuthZ ticket format for extended AuthZ session management
- Summary and Future developments
- Additional materials
  - XACML policy examples

GAAA – Generic Authentication, Authorization, Accounting

GAAA-AuthZ – GAAA AuthZ Framework

Background - Target projects

- AuthZ service for dynamic (distributed) Grid applications
  - Adding extended security context management to Grid oriented AuthZ Frameworks (EGEE gJAF and Globus GT-AuthZ)
- Distributed multidomain Authorisation service for network on-demand services and OLPP
  - EU Project PHOSPHORUS and NL national project RoN GP-NG
    - Requires extended AuthZ/provisioning session context management in multidomain scenario
- Central Authorisation service for Grid based Collaborative applications
  - GAAA-AuthZ Architecture and Implementation (Collaboratory.nl, VL-e projects)
    - Implements Domain based resource management and RBAC (RBAC-DM)
- Std framework - OGSA-AUTHZ WG at Open Grid Forum
  - Grid AuthZ service components
  - AuthZ session management

Background – Generic AuthZ Components and Mechanisms

- An "authorization" is a process by which a right or a permission is granted to an entity/subject to access a resource.
- AuthZ Service Components
  - Subject (ID, Attrs), Policy (Locality/Environment), Resource/Object (State)
- AuthZ service interoperation and compatibility
  - The same AuthZ decision on the same set of Subject attributes based on the same Resource state
    - May contain Conditions/Obligations implied by the Policy decision
  - *Example 1: The same tour booked via different tourist offices (even if in different countries)*
- Basic mechanisms for interoperability
  - Credentials/Attributes validation/mapping
  - AuthZ decision assertions or tickets (usually bound to AuthZ session)
  - Authority binding (to convey trust relations)
    - All credentials and policy should match authority/issuer

AuthZ Models and Frameworks

AuthZ service component models

- User/AuthZ session and attributes management – RBAC, ITU/ISO X.812 PMI, GAAA-AuthZ, AAI, Shibboleth
- Application integration – Interceptor/Axis model (gJAF, GT4-AuthZ, Acegi), generic AAA-API
- Policy type – BlackList, ACL, gridmap, XACML, PERMIS
- Credentials/Attributes – X.509 AC/VOMS , SAML, Shibboleth

Existing AuthZ frameworks

- EGEE gLite Java AuthZ Framework and Globus GT-AuthZ
- LCAS/LCMAPS
- PERMIS
- GAAA-AuthZ (by UvA)
- COPS (Common Open Policy Service) – RFC2748, RFC2753, RFC3761
- Acegi (for J2EE/Spring)
- Shibboleth, Liberty and A-Select based AAI

Complex Resource Provisioning (CRP)

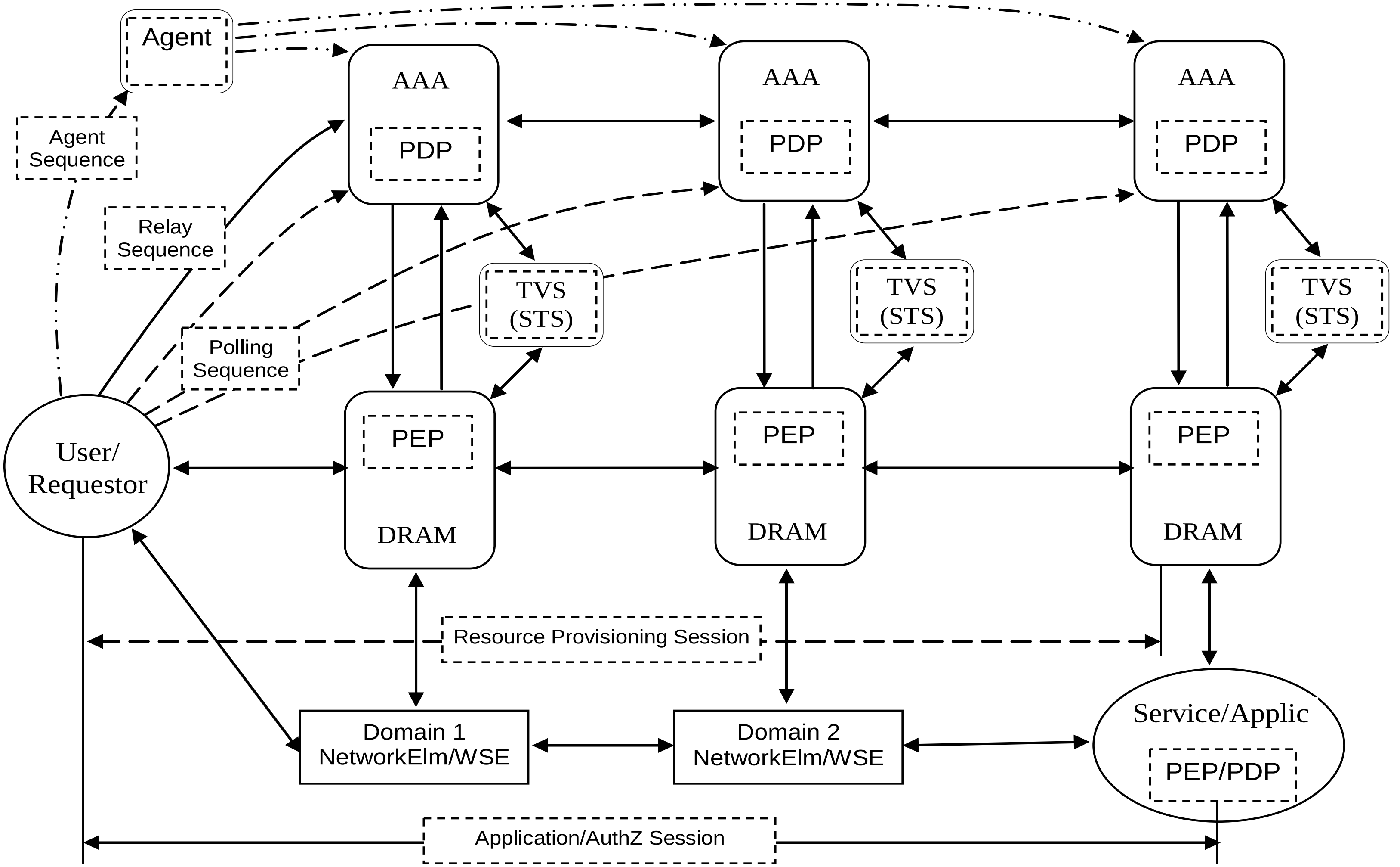
Basic use cases for CRP

- *OLPP and Network on-demand provisioning*
- *Virtual Laboratory - Hierarchical and distributed resources and user attributes*
- *Grid Computing Resource – Virtualised, distributed and heterogeneous*

2 major stages/phases in CRP operation

- *Provisioning consisting of 4 basic steps*
    - *Resource Lookup*
    - *Resource composition (including options)*
    - *Component resources reservation (reservation ID)*
    - *Deployment*
- *Access (to the resource) or consumption (of the consumable resource)*
    - *TBN reservation/AuthZ decision enforcement*

CRP infrastructure elements and basic sequences



Provisioning sequences

* Polling

* Relay

* Agent

TVS – Token Validation Service

DRAM – Dynamic Resource Allocation and Mngnt

PDP – Policy Decision Point

PEP – Policy Enforcement Point

Required AAA/Service plane functionality for OLPP/CRP

Authentication and Identity management

- Federated Identity and Federated Resource Access
- Attribute management (issue, validation, mapping, delegation)

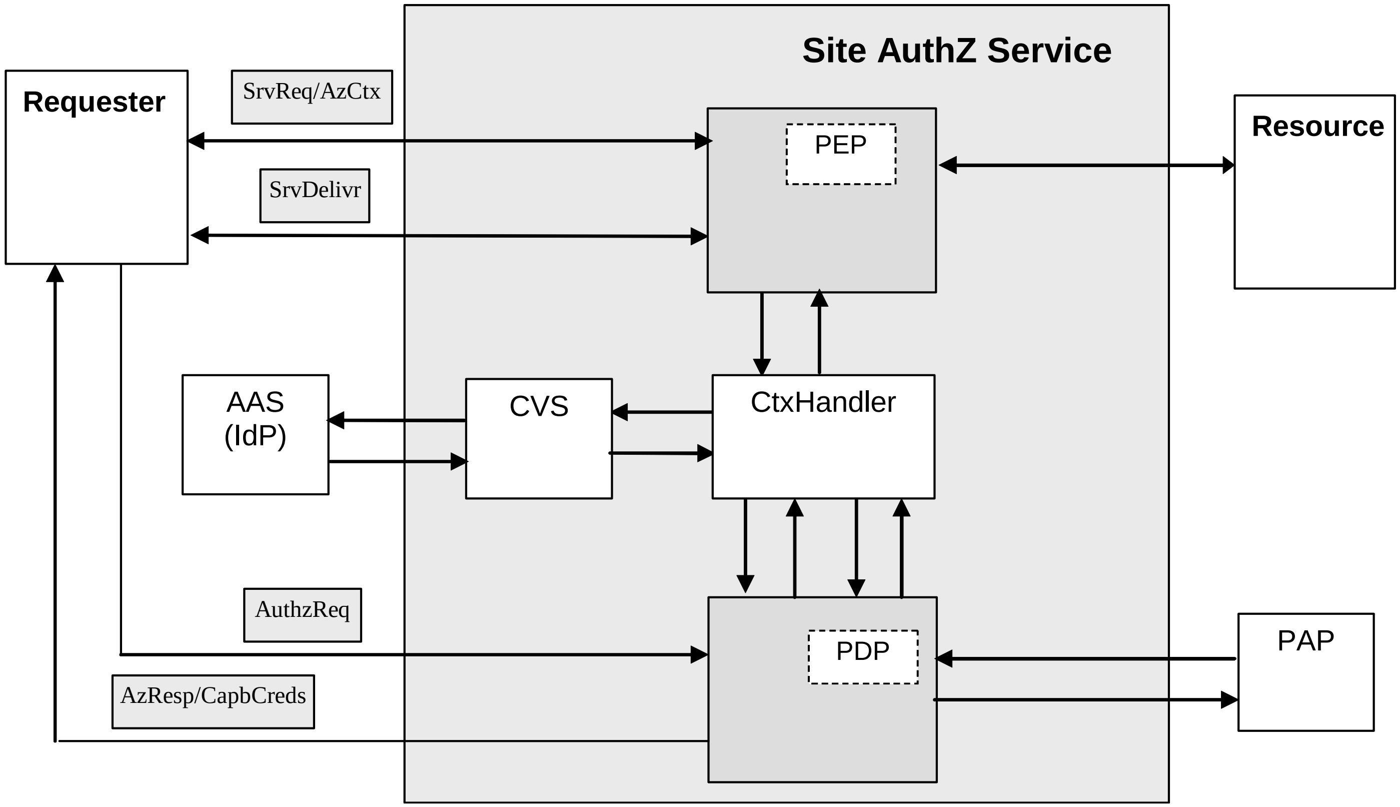
Authorisation

- Multidomain AuthZ policy and/or decisions combination
- AuthZ session Management to convey AuthZ decision between domains

Trust management

- User and Resource based Federations (Shibboleth, NREN/GN2 AAI, VO)
  - ◆ Pre-established trust relations
- Trusted Computing Platform (TCG)
  - ◆ Hardware rooted trust anchors allowing for initial trusted introduction
- DNSSEC
  - ◆ Allows for DNS based VO certificates publishing to enable initial trusted introduction

AuthZ Service Components (OGSA-AUTHZ)



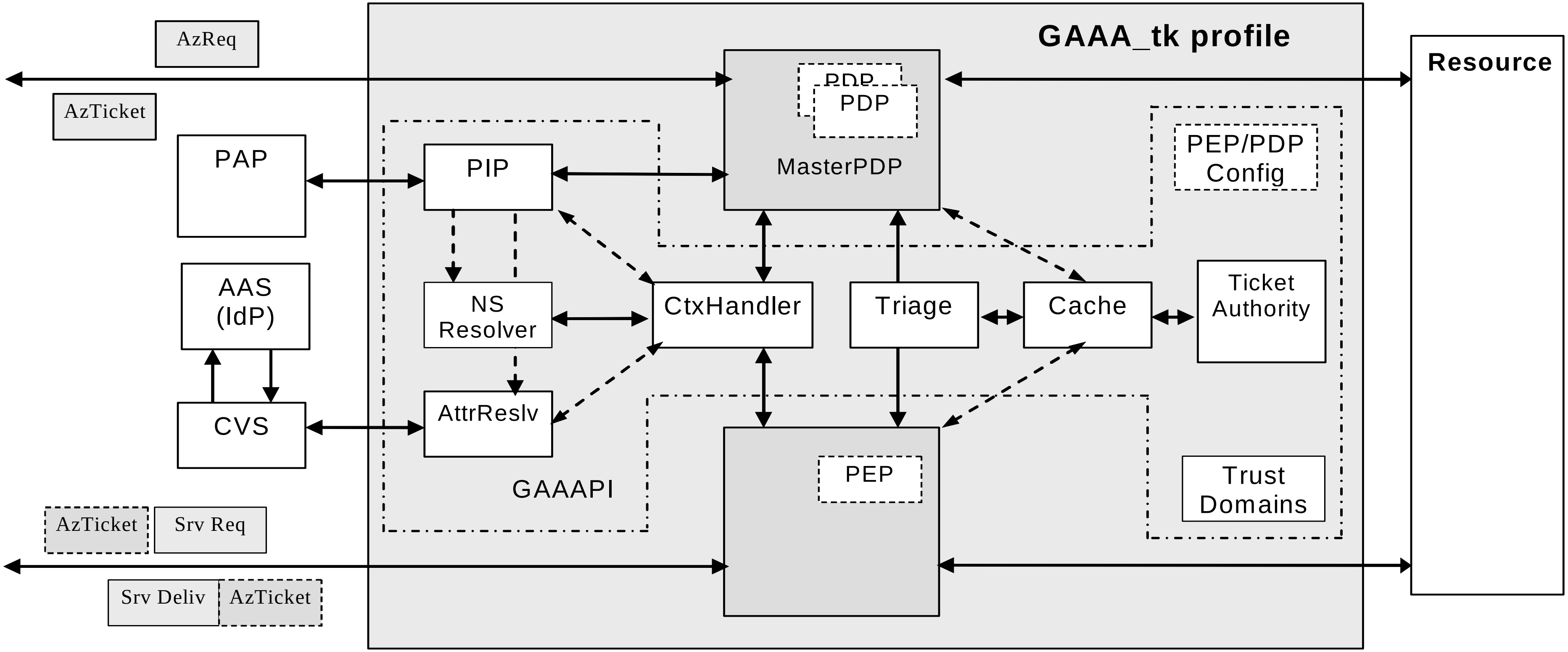
Basic sequences for attributes providing and for AuthZ decisions

- Push
- Pull
- Agent

CVS – Credentials Validation Service

PAP – Policy Authority Point

GAAA-AuthZ/GAAAPI components to support dynamic security context management (1)



- GAAAPI is a collection of components to support PEP and PDP interaction, implemented in Java
- Needs Trust Anchor configuration in a distributed multidomain infrastructure

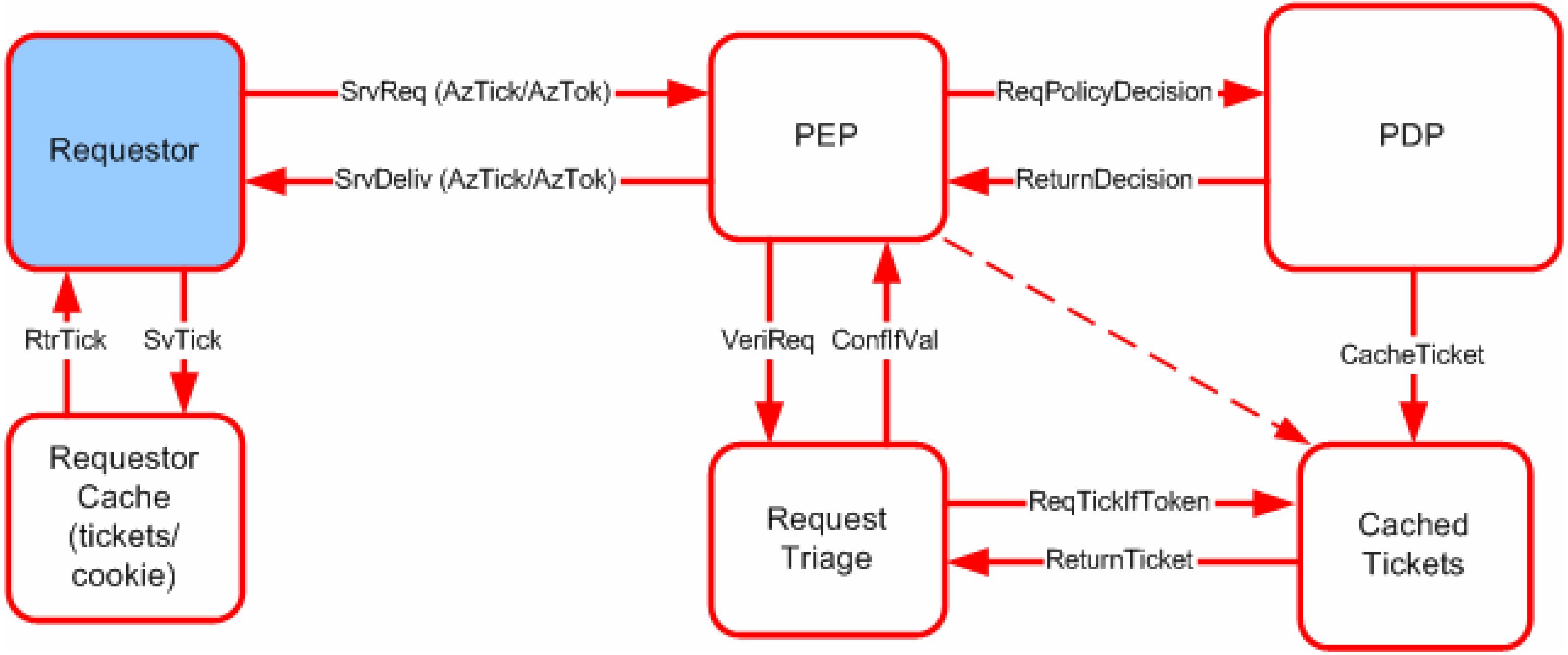
GAAAPI components to support dynamic security context management (2)

- Context Handler (CtxHandler) that calls to a namespace resolver (NS Resolver) and attribute resolver (AttrResolver), which in its own can call to external CVS or Attribute Authority Service (AAS) *to validate* presented attributes or obtain new ones
- Triage and Cache to provide an initial evaluation of the request, including the validity of the provided credentials
  - Used for handling AuthZ tickets/tokens, and also for AuthZ session management by evaluating service requests versus the provided AuthZ ticket/token claims
- Ticket Authority (TickAuth) generates and validates AuthZ tickets or tokens on the requests from PEP or PDP
  - to support AuthZ session, tickets are cached by TickAuth directly or by PEP/PDP
- Policy Information Point (PIP) that provides resolution and call-outs to related authoritative Policy Authority Points (PAP)

AuthZ Session management in GAAA-AuthZ

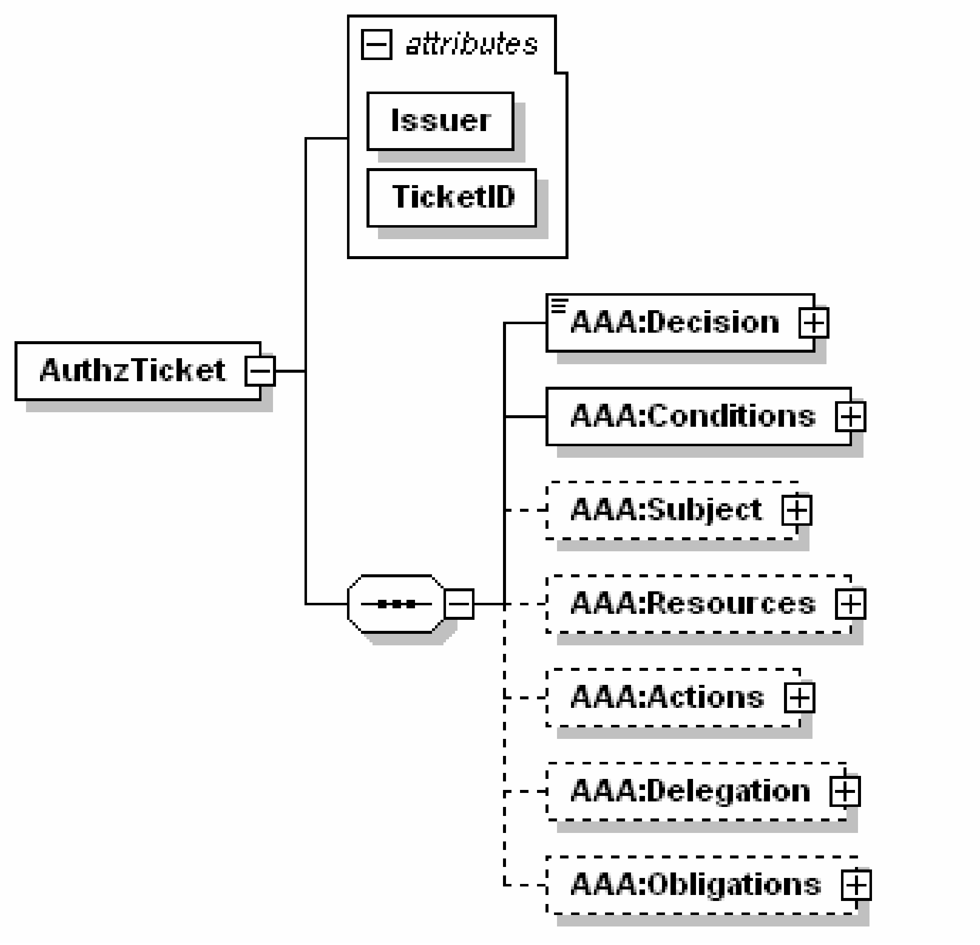
- AuthZ session is a part of the generic GAAA-AuthZ (and RBAC) functionality
- Session can be started only by an authorised Subject/Role
  - Session can be joined by other less privileged users
  - Session permissions/credentials can be delegated to (subordinate) subjects
- Session context includes Request/Decision information and may include any other environment or process data/information
  - AuthZ Session context is communicated in a form of extended AuthZ Assertion or AuthZ Ticket
  - SessionID is included into AuthzTicket together with other AuthZ Ctx information
  - Signed AuthzTicket is cached by PEP or PDP
- If session is terminated, cached AuthzTicket is deleted
  - Note: AuthzTicket revocation should be done globally for the AuthZ trust domain

AuthZ session Tickets/Tokens handling in AuthZ system



- AuthzTicket is issued by PDP and may be issued by PEP
- AuthzTicket must be signed
- AuthzTicket contains all necessary information to make local PEP-Triage Request verification
- When using AuthzTokens, AuthzTickets must be cached; Resolution mechanism from token to ticket must be provided

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

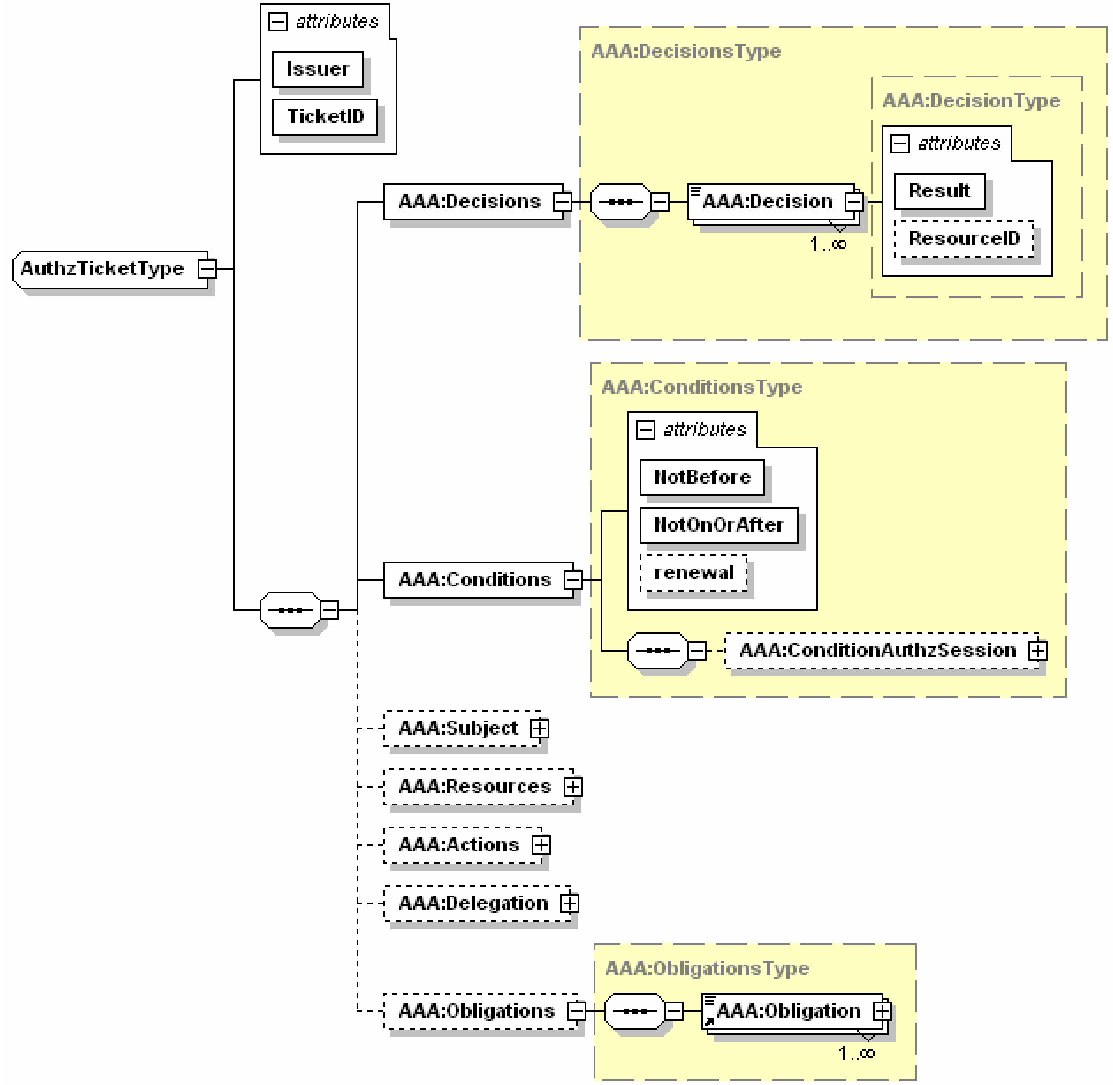
Ensure Integrity of the AuthZ decision

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

Confidentiality

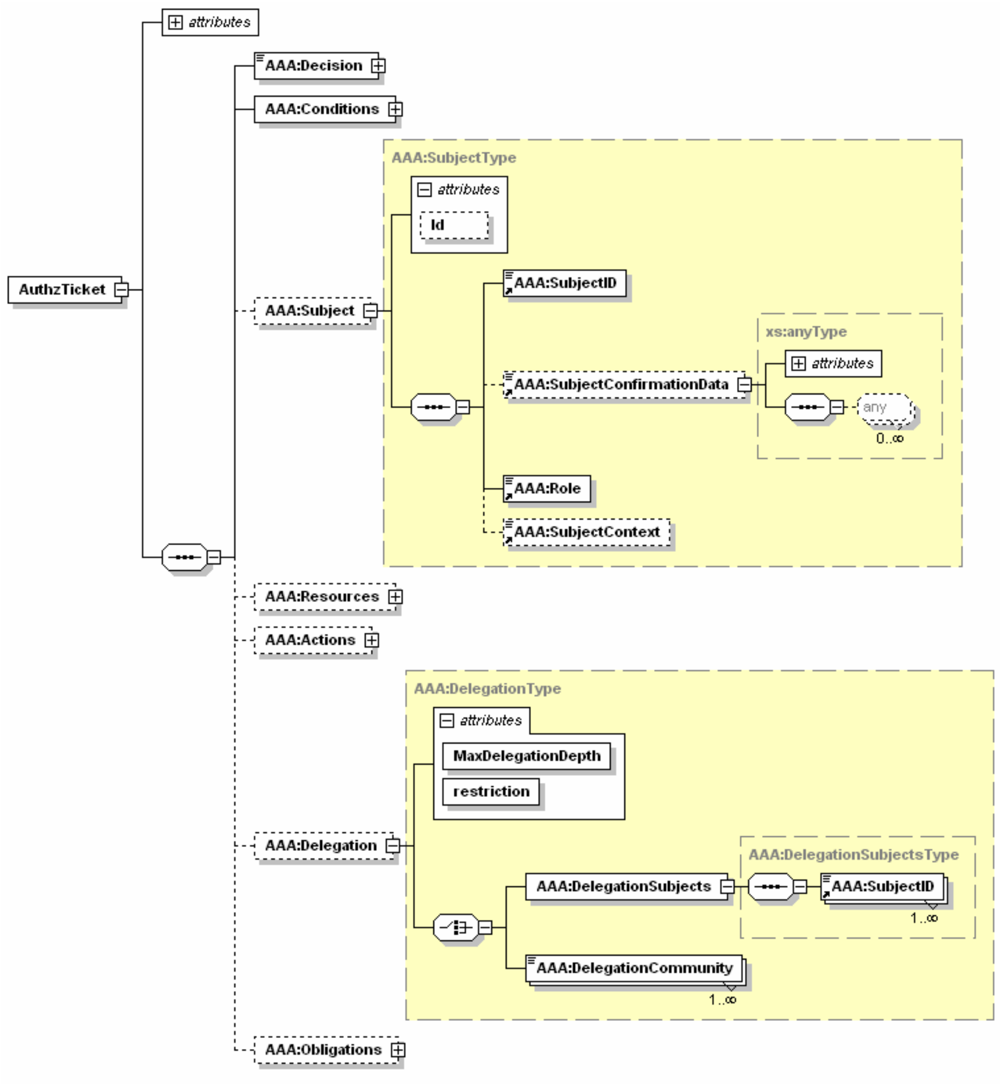
- Creates a basis for user-controlled Secure session

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>
```

AuthzToken example – 293 bytes

```
<AAA:AuthzToken TokenID="c24d2c7dba476041b7853e63689193ad">
<AAA:TokenValue>
0IZt9WsJT6an+tIxhhTPtiztDpZ+iynx7K7X2Cxd2iBwCUTQ0n61Szv81DKllWsq75IsHfusnm56
zT3fhKU1zEUsob7p6oMLM7hb42+vjfvNeJu2roknhIDzruMrr6hMDsIfaotURepu7QCT0sADm9If
X89Et55EkSE9oE9qBD8=
</AAA:TokenValue>
</AAA:AuthzToken>
```

AuthzToken is constructed of the AuthzTicket TicketID and SignatureValue

AuthzToken use suggests caching AuthzTicket’s

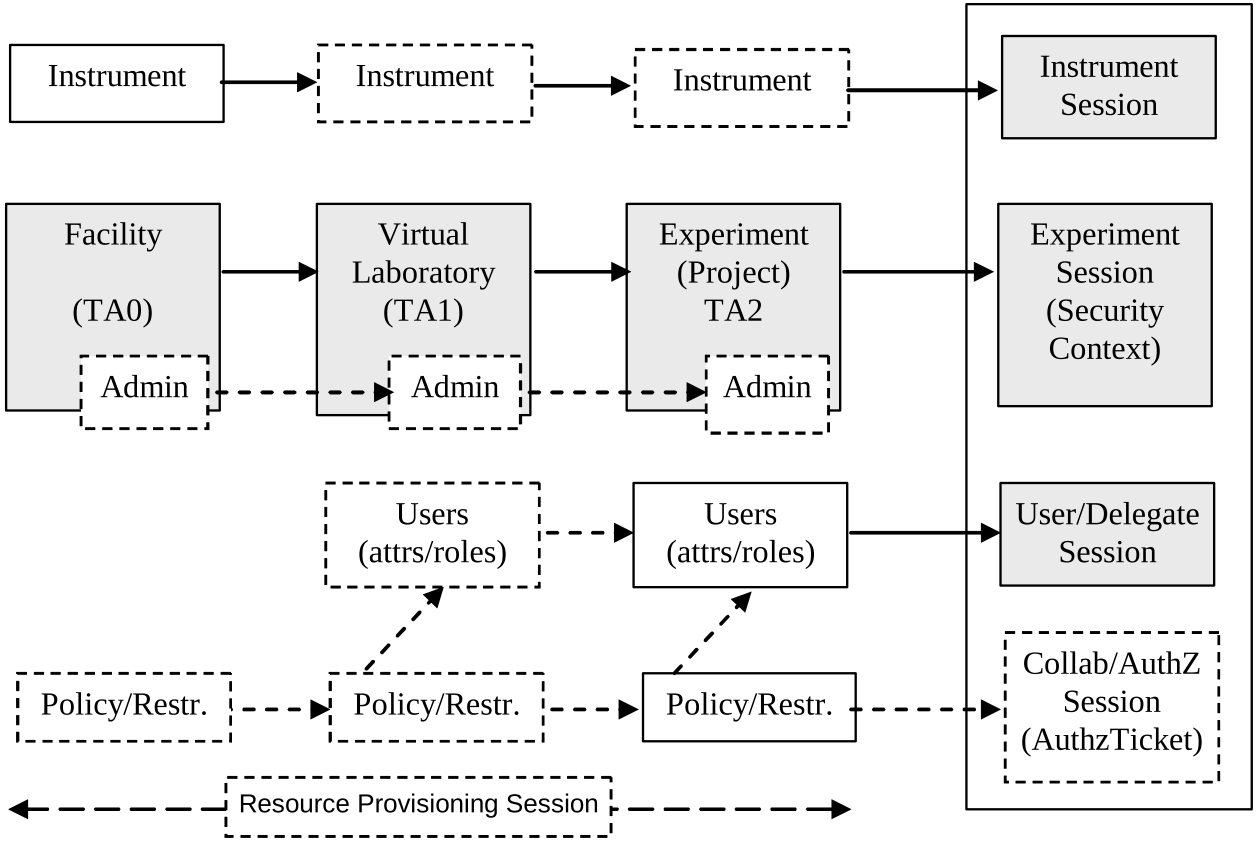
Future developments

- Proposing AuthZ session management framework to OGSA-AUTHZ
- AuthZ session management with the extended AuthZ ticket functionality
  - Including delegation and complex and obligated policy decisions
  - Needs more discussion on Delegation use cases and scenarios
- *Dynamic Trust management in multidomain CRP*
- Defining XACML policy profiles for
  - Grid applications (mapping between Grid specific policy formats – gridmap, ACL, GACL)
  - different Resource models (hierarchical, ordered, mesh, etc.)
- Integration with existing access control frameworks GT4-AuthZ, gJAF, Acegi

Additional information

- Complex AuthZ scenarios – Use cases
  - Hierarchical Domain based Resource management in GCE
  - Complex Resource Provisioning (on-demand) – OLPP example
  - User Controlled Virtual Workspace Service (VWSS-UC)
- XACML policy examples

Domain based Resource management



Implements RBAC3 model + Experiment AuthZ session management

Uses XACML RBAC profile and XACML v3.0 administrative policy profile

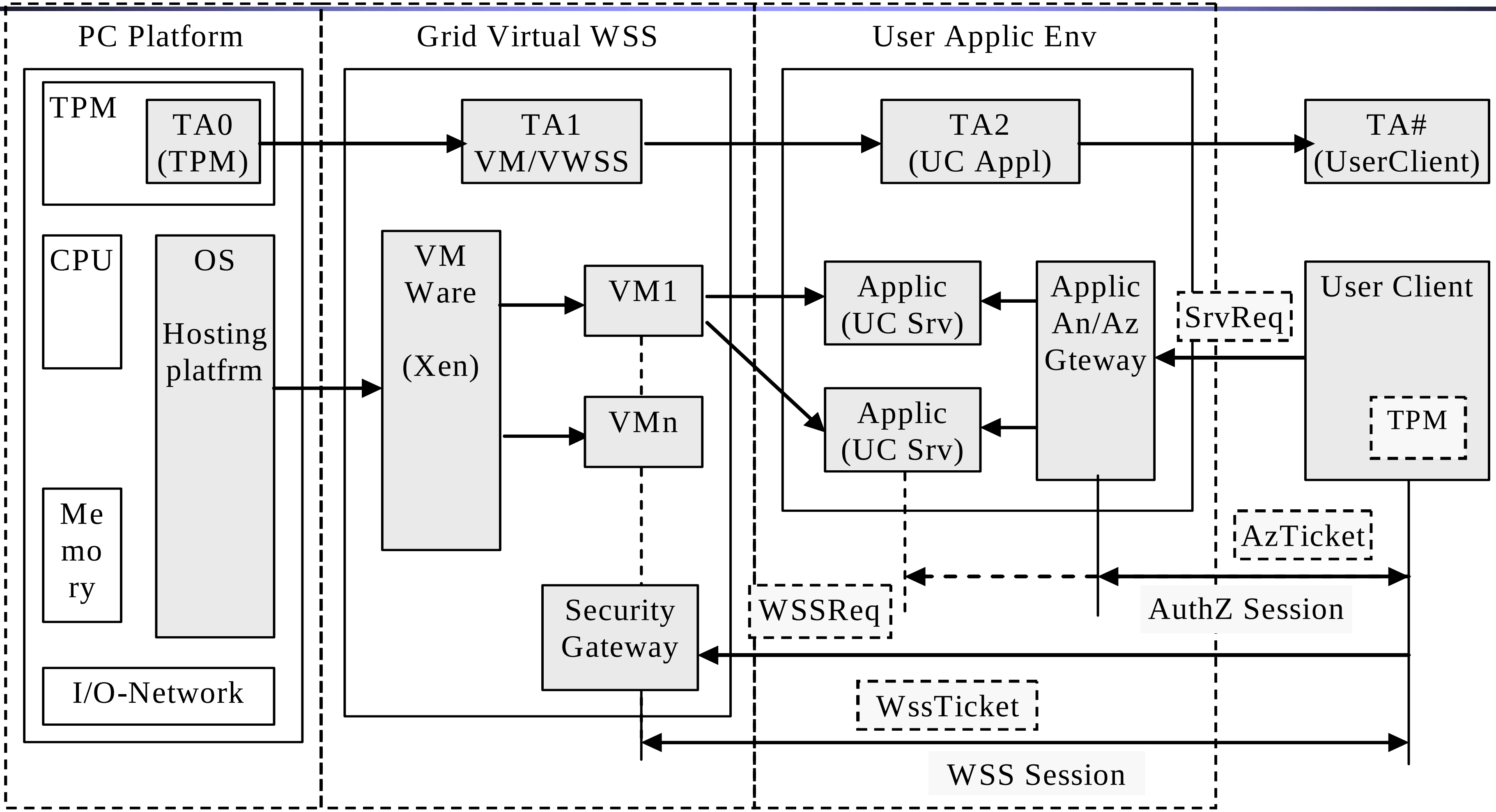
Full Resource URI/ID –

CNL:Facility:VirtualLab:Experiment:InstrModel

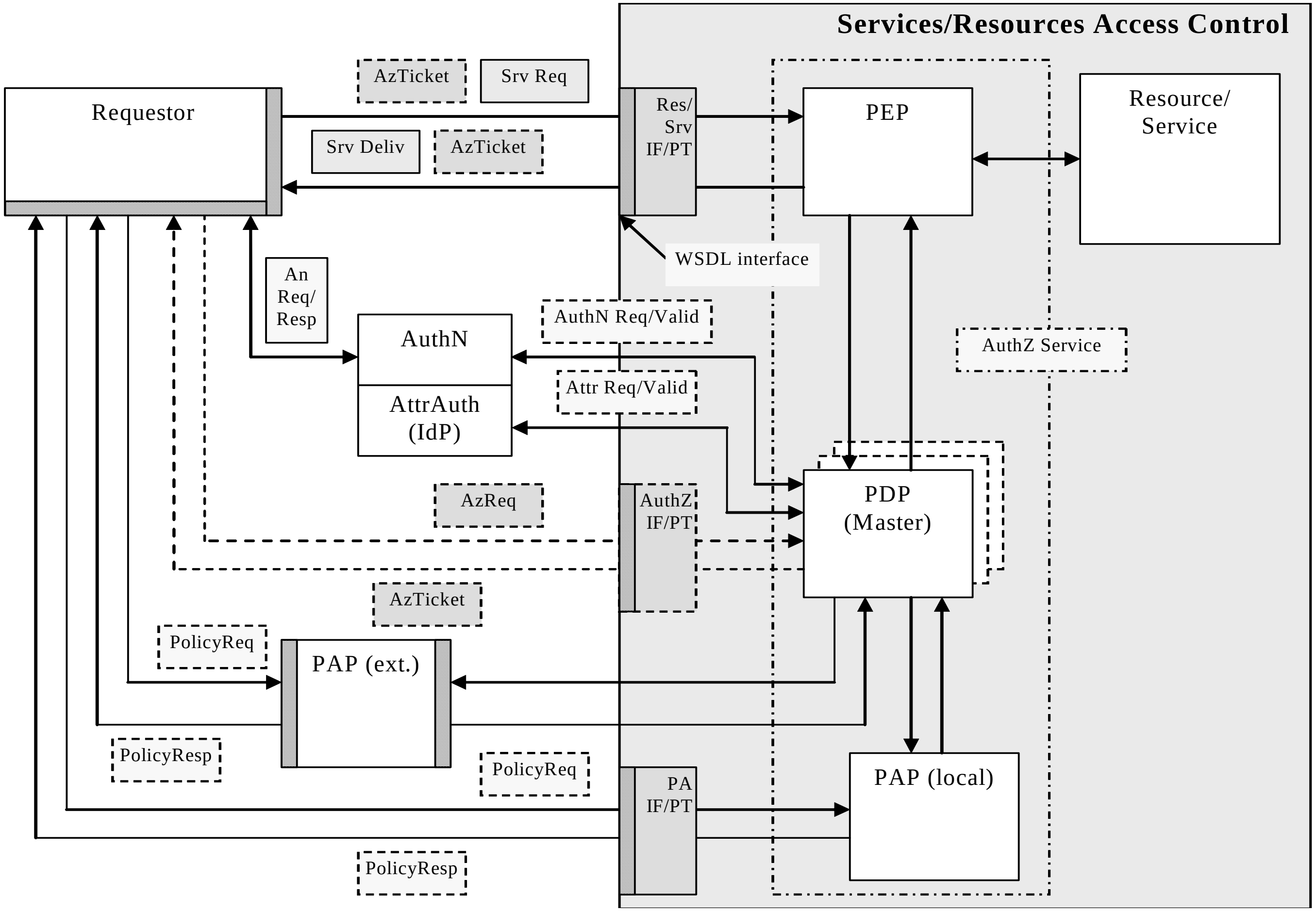
Full User Session context –

Facility < Virtual Lab < Experiment < Experiment Session < Collaborative Session

User Controlled Virtual Workspace Service (VWSS-UC)



AuthZ service operation in Grid/WS based applications



Security/Access control services integrated with the Workflow via Web Services ports and messages definition

Message-level Security services are linked to SOAP header

Linking dynamically all components of the access control system

Policy is attached to any component of the service description in WSDL format

Interacting services will fetch policy document and apply restrictions/rules to elements, which declared policy compliance requirements

Security context management:
Context dependent information and existing mechanisms

Context dependent information/attributes:

- Policy
- Trust domains and authorities
- Attributes namespaces
- Service/Resource environment/domain
- Credential semantics and formats

Mechanisms to transfer/manage context related information

- Service and requestor/user ID/DN format that should allow for both using namespaces and context aware names semantics
- Attribute format (either X.509/X.521 or URN/SAML2.0 presentation)
- Context aware XACML policy definition using the Environment element of the policy Target element
- Security tickets and tokens used for AuthZ session management and for provisioned resource/service identification
- Security federations for users and resources, e.g. VO membership credentials

GAAA-RBAC/GAAAPI Security Configuration

General security configuration

- Key store location and access
- Trusted and local keys/credentials

Trust domains and authorities (depending on possible PEP and PDP location)

- PEP is protecting Resource, and therefore should be located in the Resource trust domain
- PDP may be remote, in this case communication between PEP and PDP must be protected cryptographically

PEP and PDP Configuration (at invocation time)

- Namespace Resolver
- AuthzTicket Authority (tickauthPDP | tickauthPEP)
- Trust domains
- Session credentials or AUthZ ticket/tokens format

PDP Configuration

- Standard XACML PDP configuration
- *In development:* Master PDP configuration with components and Request/policy evaluation (micro)flow

XACML Special profiles for RBAC and complex Resources

XACML RBAC profile

- defines policies that require multiple Subjects and roles combination to access a resource and perform an action
- implements hierarchical RBAC model when some actions require superior subject/role approval to perform a specific action
- can significantly simplify rights delegation inside the group of collaborating entities/subjects

XACML Hierarchical Resource profile

- defines policy format for hierarchically organised resources, e.g. file system or XML-based repositories

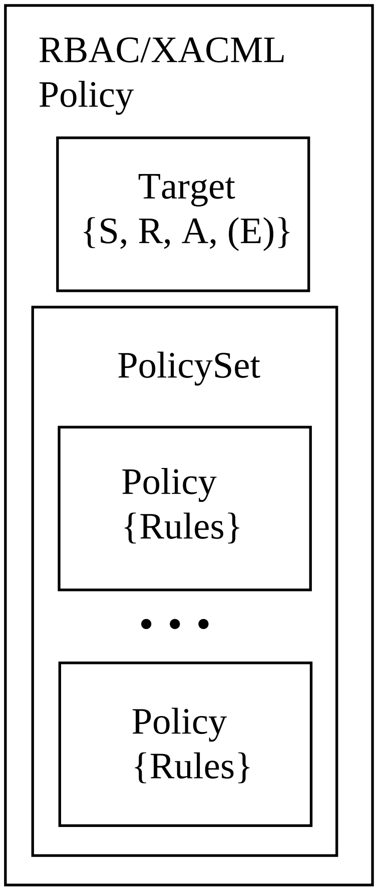
XACML complex Resource profile

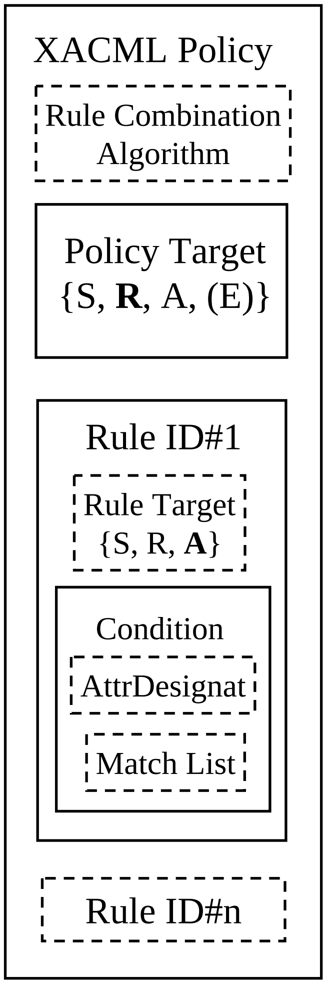
- allows for complex request to multiple resources having the same request context, however decision is provided per resource

XACML3 Delegation profile

XACML Policy structure

XACML Policy format





CNL AuthZ policy: XACML Policy generation conventions

- Policy Target is defined for the Resource
- Policy combination algorithm is “ordered-deny-override” or “deny-override”
- Rule Target is defined for the Action and may include Environment checking
  - Rule’s Condition provides matching of roles which are allowed to perform the Action
- Access rules evaluation
  - Rules are expressed as permissions to perform an action against Subject role
  - Rule combination algorithm “permit-override”
  - Rules effect is “Permit”
- Subject and Credentials validation – is not supported by current XACML functionality
  - Credential Validation Service (CVS) – proposed GGF-AuthZ WG development

RBAC AuthZ policy: Resource, Actions, Subject, Roles

Actions (8)

- StartSession
- StopSession
- JoinSession
- ControlExperiment
- ControlInstrument
- ViewExperiment
- ViewArchive
- AdminTask

Roles (4)

- Analyst
- Customer
- Guest
- Administrator
- (CertifiedAnalyst)

Naming convention

- Resource - “http://resources.collaboratory.nl/Phillips_XPS1”
- Subject – “WHO740@users.collaboratory.nl”
- Roles - “role“ or “role@ExperimentID”

Simple Access Control table

Roles	Anlyst	Custm	Guest	Admin
ContrExp	1	0	0	0
ContrInstr	1	0	0	1
ViewExp	1	1	1	0
ViewArch	1	1	0	1
AdminTsk	0	0	0	1
StartSession	1	0	0	0
StopSession	1	0	0	1
JoinSession	1	1	1	0

See XACML policy example =>

```
<Policy PolicyId="urn:oasis:names:tc:xacml:1.0:cnl2:policy:CNL2-XPS1" RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-
                algorithm:deny-overrides">
 <Description>Permit access for CNL2 users with specific roles</Description>
 <Target>
  <Subjects>
   <AnySubject/>
  </Subjects>
  <Resources>
   <Resource>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:anyURI-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#anyURI">http://resources.collaboratory.nl/Phillips_XPS1</AttributeValue>
     <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
                DataType="http://www.w3.org/2001/XMLSchema#anyURI"/>
    </ResourceMatch>
   </Resource>
  </Resources>
  <Actions>
   <AnyAction/>
  </Actions>
 </Target>
 <Rule RuleId="urn:oasis:names:tc:xacml:1.0:urn:cnl:policy:urn:oasis:names:tc:xacml:1.0:cnl2:policy:CNL2-XPS1:rule:ContrExp"
                Effect="Permit">
  <Target>
   <Subjects>
    <AnySubject/>
   </Subjects>
   <Resources>
    <AnyResource/>
   </Resources>
   <Actions>
    <Action>
     <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">ContrExp</AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ActionMatch>
    </Action>
   </Actions>
  </Target>
  <Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">analyst</AttributeValue>
   </Apply>
   <SubjectAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:subject:role" DataType="http://www.w3.org/2001/XMLSchema#string"
                Issuer="CNL2AttributeIssuer"/>
  </Condition>
 </Rule>
 <Rule RuleId="urn:oasis:names:tc:xacml:1.0:urn:cnl:policy:urn:oasis:names:tc:xacml:1.0:cnl2:policy:CNL2-XPS1:rule:ContrInstr"
                Effect="Permit">
  <Target>
   <Subjects>
    <AnySubject/>
   </Subjects>
   <Resources>
    <AnyResource/>
   </Resources>
   <Actions>
    <Action>
     <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">ContrInstr</AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ActionMatch>
    </Action>
   </Actions>
  </Target>
  <Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">analyst</AttributeValue>
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">admin</AttributeValue>
   </Apply>
   <SubjectAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:subject:role" DataType="http://www.w3.org/2001/XMLSchema#string"
                Issuer="CNL2AttributeIssuer"/>
  </Condition>
 </Rule>
 <Rule RuleId="urn:oasis:names:tc:xacml:1.0:urn:cnl:policy:urn:oasis:names:tc:xacml:1.0:cnl2:policy:CNL2-XPS1:rule:ViewExp"
                Effect="Permit">
  <Target>
   <Subjects>
    <AnySubject/>
   </Subjects>
   <Resources>
    <AnyResource/>
   </Resources>
   <Actions>
    <Action>
     <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">ViewExp</AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ActionMatch>
    </Action>
   </Actions>
  </Target>
  <Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">analyst</AttributeValue>
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">customer</AttributeValue>
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">guest</AttributeValue>
   </Apply>
```

GT4 Authorisation Framework

Can be configured for Container, Message, Service/Resource

- Called from the SOAP/Axis message interceptor

AuthZ processing sequence includes

- Bootstrapping X.509 PIP – retrieves request parameters from the message
  - Subject, Resource, Action
- Sequence of pre-configured PIP’s, including SAML
- Sequence of (specialised) PDP’s
- Different PDP decisions combination algorithms by AuthZ engine
  - However, multiple policy decision’s consistency is not resolved

Available PDP’s

- ACL and GridMap
- HostAuthorization and UserNameAuthorization (similar BlackList PDP)
- SAML AuthZ callout and SAML AuthZ Assertion
- SelfAuthorization – based on shared/trusted Resource credentials
- Simple XACML PDP (provided as a placeholder for extension)

gJAF Overview – Current Design

Provided as org.glite.security.authz Java package

- Uses actively org.glite.security.utils
- Has inherited (architectural) compatibility with GT4-AuthZ

Called from applications via an interceptor/gateway

- {MessageContext, Subject, operation}

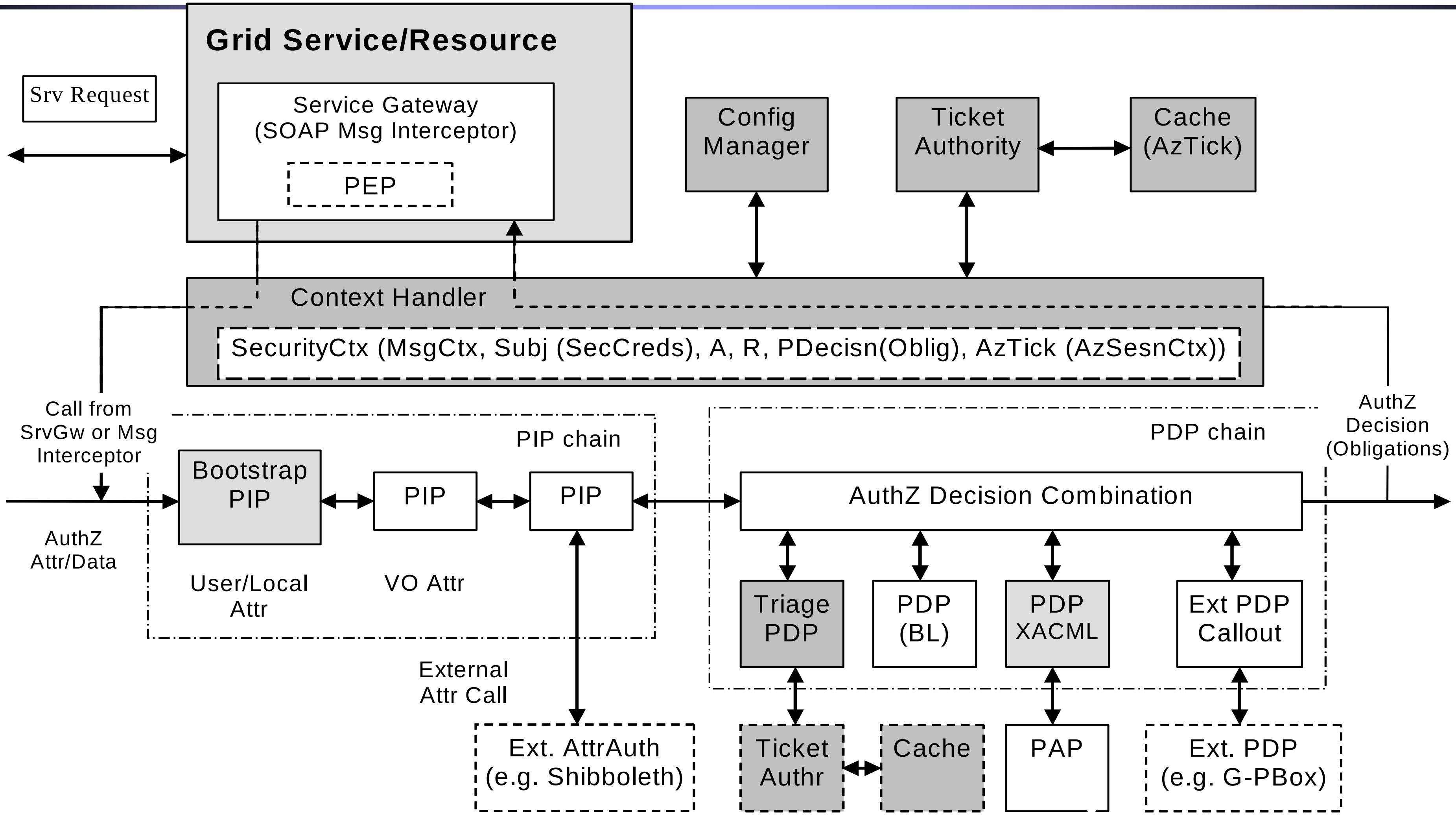
Contains a configured chain of PIP and PDP modules

- PIP collects/extracts information to be sent to PDP
- Each PDP evaluates its relevant attributes against its own Policy
- Chain is configured to apply PDP decisions combination

Problems

- Requires application specific manual chain configuration
- Limited use up to now in gLite by CREAM

gJAF – Proposed Extensions


Grid Service/Resource
Srv Request
Service Gateway
(SOAP Msg Interceptor)
PEP
Config
Manager
Ticket
Authority
Cache
(AzTick)
Context Handler
SecurityCtx (MsgCtx, Subj (SecCreds), A, R, PDecisn(Oblig), AzTick (AzSesnCtx))
Call from
SrvGw or Msg
Interceptor
PIP chain
Bootstrap
PIP
PIP
PIP
AuthZ
Attr/Data
User/Local
Attr
VO Attr
External
Attr Call
Ext. AttrAuth
(e.g. Shibboleth)
PDP chain
AuthZ
Decision
(Obligations)
AuthZ Decision Combination
Triage
PDP
PDP
(BL)
PDP
XACML
Ext PDP
Callout
Ticket
Authr
Cache
PAP
Ext. PDP
(e.g. G-PBox)