Authorisation service coordination: EGEE internal and inter-project

Yuri Demchenko
University of Amsterdam

MWSG10 meeting
November 14-15, 2006, CERN

www.eu-egee.org

Information Society and Media

Outline

- **AuthZ coordination and interoperability**
  - Abstraction
  - Models and Frameworks
  - Initiatives and activities
- **gJAF Overview and further development**
- **Discussion – intra-EGEE and inter-project coordination**

Abstraction - AuthZ Components and Mechanisms

- **AuthZ service interoperation and compatibility**
  - Components: Subject (ID, Attrs), Policy (Locality/Environment), Resource (State)
  - The same AuthZ decision on the same set of Subject attributes based on the same Resource state
    - May contain Conditions/Obligations
  - *Example 1: The same tour booked via different tourist offices (even if in different countries)*
  - *Example 2: Diplomatic passport (or "Bourn Identity")*
- **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, PMI, Shibboleth
  - Application integration – Interceptor/Axis model (gJAF, GT4-AuthZ, Acegi), GAAA-API
  - Policy type – BlackList, ACL, gridmap, XACML, PERMIS
  - Credentials/Attributes – X.509 AC/VOMS , SAML, Shibboleth

- **Existing AuthZ frameworks**
  - gLite Java AuthZ Framework and GT4-AuthZ
  - LCAS/LCMAPS (is a Framework?)
  - PERMIS
  - GAAA-AuthZ
  - Acegi (for J2EE/Spring)
  - Shibboleth based?

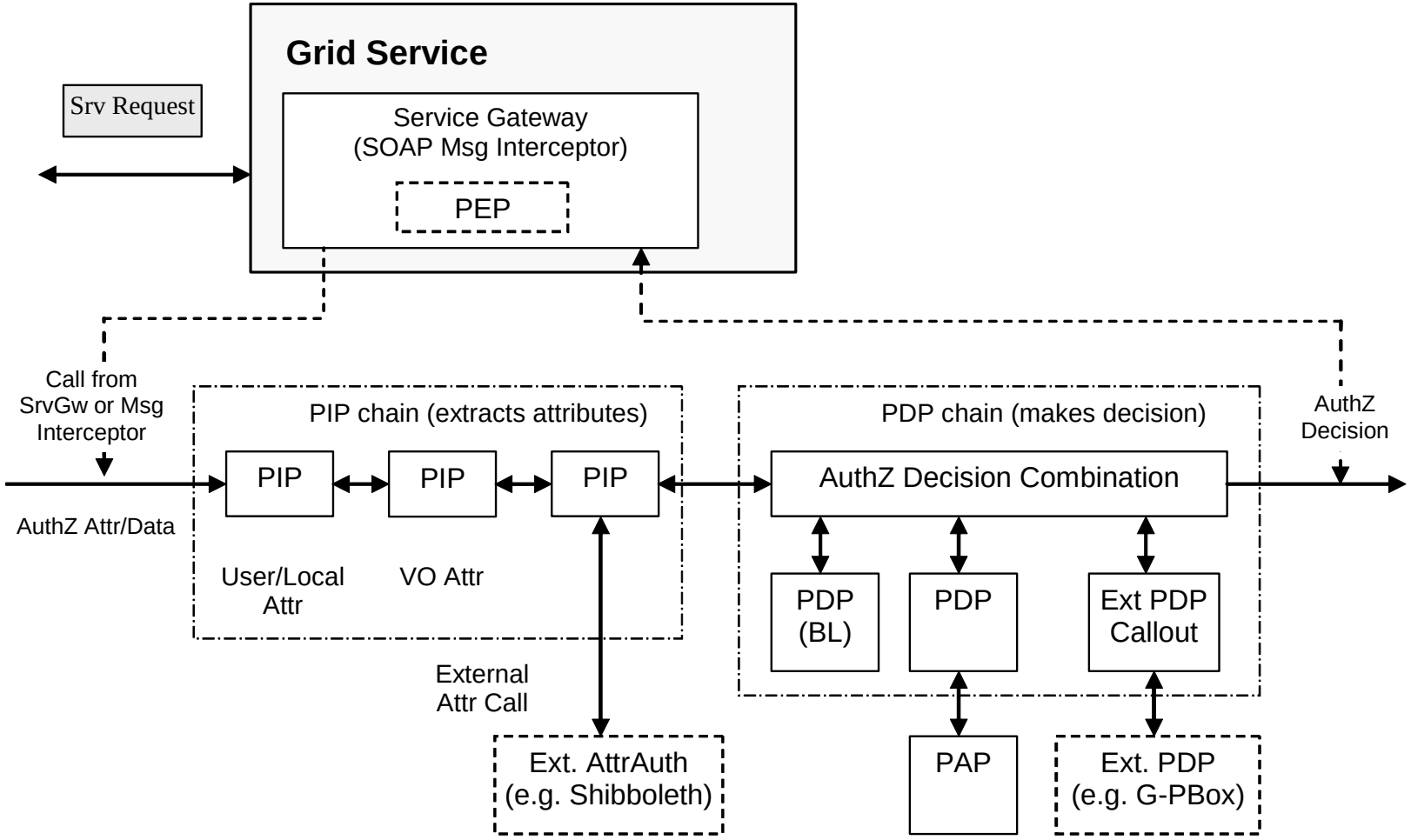
AuthZ Coordination Initiatives

- **OGSA-AUTHZ (**https://forge.gridforum.org/projects/ogsa-authz**) and other OGF initiatives**
  - Functional Components of Grid Service Provider Authorisation Service Middleware
  - Credential Validation Service (CVS)
    - Implements one of mechanism for interoperability
  - WS-TRUST and SAML to access a CVS and Request Context to access a PDP
  - CVS Requirements – not formal
- **Interoperation and integration with campuses**
  - Accepting Shibboleth attributes
    - VOMS-Shibboleth integration – GridShib, GridAAI (by SWITCH :-))

gJAF Overview

- **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 components and connection to the Grid Service



Suggested further development (1)

- **SAML/Shib Credentials support**
  - To be provided as internal gJAF package or part of org.glite.security.utils and supported by SAML/Shib PIP
    - Need to clarify SAML Assertions format and supporting libraries
    - Will rely on effective cooperation with SWITCH
  - Also expected to be available in GT4-AuthZ with GridShib
- **Using XACML for policy expression**
  - Motivation - Standard, Context aware, can be mapped to different formats
    - Used in G-PBox
  - Can be added as XACML PDP plugin to gJAF or GT4-AuthZ
  - Need policy management tool (simple or complex)
- **Other issues found important**
  - Enable PDP chain to respond with Obligated decision
  - PDP answer with AuthZ ticket to provide extended/full decision context in response to gJAF/PDP

Suggested further development (2)

- **Compatibility and integration with other gLite/EGEE and 3rd party solutions**
  - Integration with the G-PBox
    - Needs gJAF AuthZ chain extension to process Obligated decisions
  - Compatibility and integration with the GT4-AuthZ
    - Possibility to reuse available set of PDP's and PIP's
    - Common interest in cooperate and compatibility with the GT4-AuthZ team
- **AuthZ Policy compatibility and coordination**
  - *Common or mapped attributes semantics*
  - Policy formats mapping – XACML, GACL, ACL, gridmap, BlackList
    - Q: Are all they compatible and convertible (e.g. to XACML)?

Other supporting activities

- **gJAF promotion in EGEE and for wider Grid community**
  - Time to update gJAF Developer's guide - https://edms.cern.ch/document/501718
  - HOWTO and usage examples
- **EGEE AuthZ Policy Coordination**
  - First meeting was in Bologna on June 6-7, 2005
  - *Time for the next meeting* – January 2007 (?)
- **OGF OGSA-AuthZ Working Group**
  - EGEE interest – bring EGEE reality to GGF standardisation
  - Proposed documents on AuthZ service components and protocols, CVS – Credentials Validation Service (to combine Shibboleth-like Attribute Service and WS-Trust-like security token service)

Discussion

- **Interaction with other packages and developers**
  - EGEE Policy coordination meeting
- **OGSA-AUTHZ**
- **Coordination with GT4-AuthZ**
- **Any other issues?**

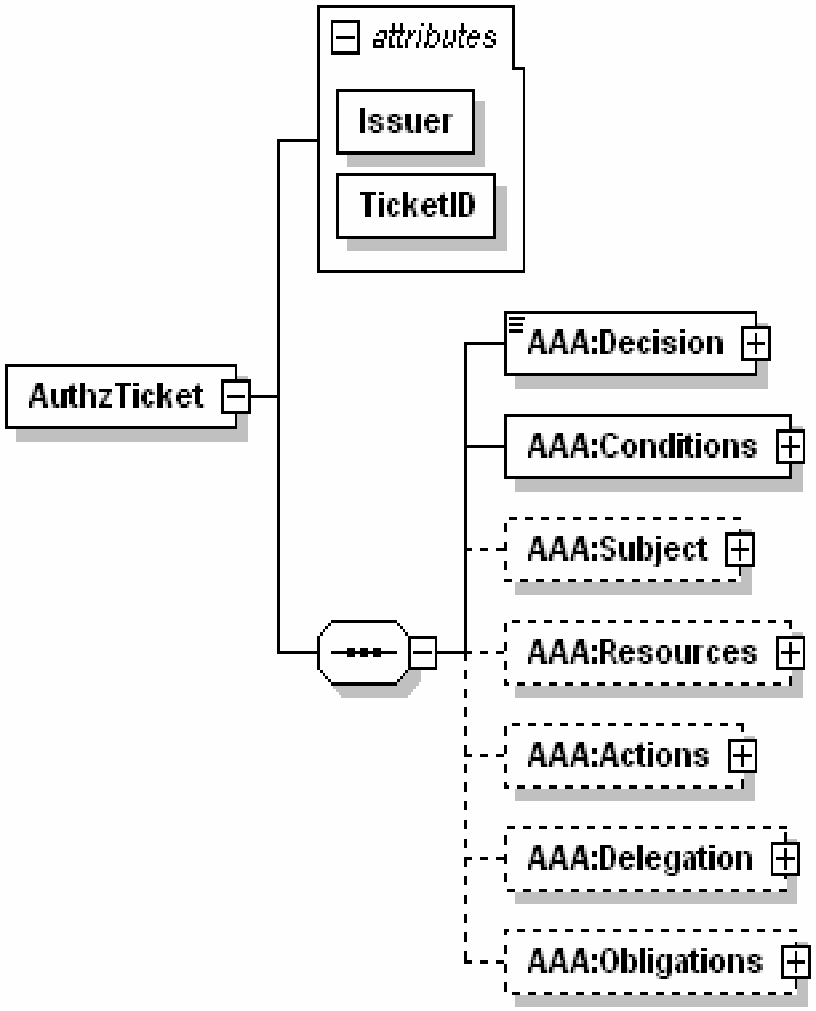
Additional information

- GT4 Authorisation Framework
- AuthZ Ticket format

GT4 Authorisation Framework

- **Can be configured for Container, Message, Service/Resource**
  - Called from the SOAP/Axis message interceptor
- **AuthZ processing sequence includes**
  - *New!* 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)

AuthZ ticket - Top elements (1)



AuthZ ticket - Top elements (2)

- `<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 Subjects or community**
  - 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 – Example (proprietary format)

```
<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>
```

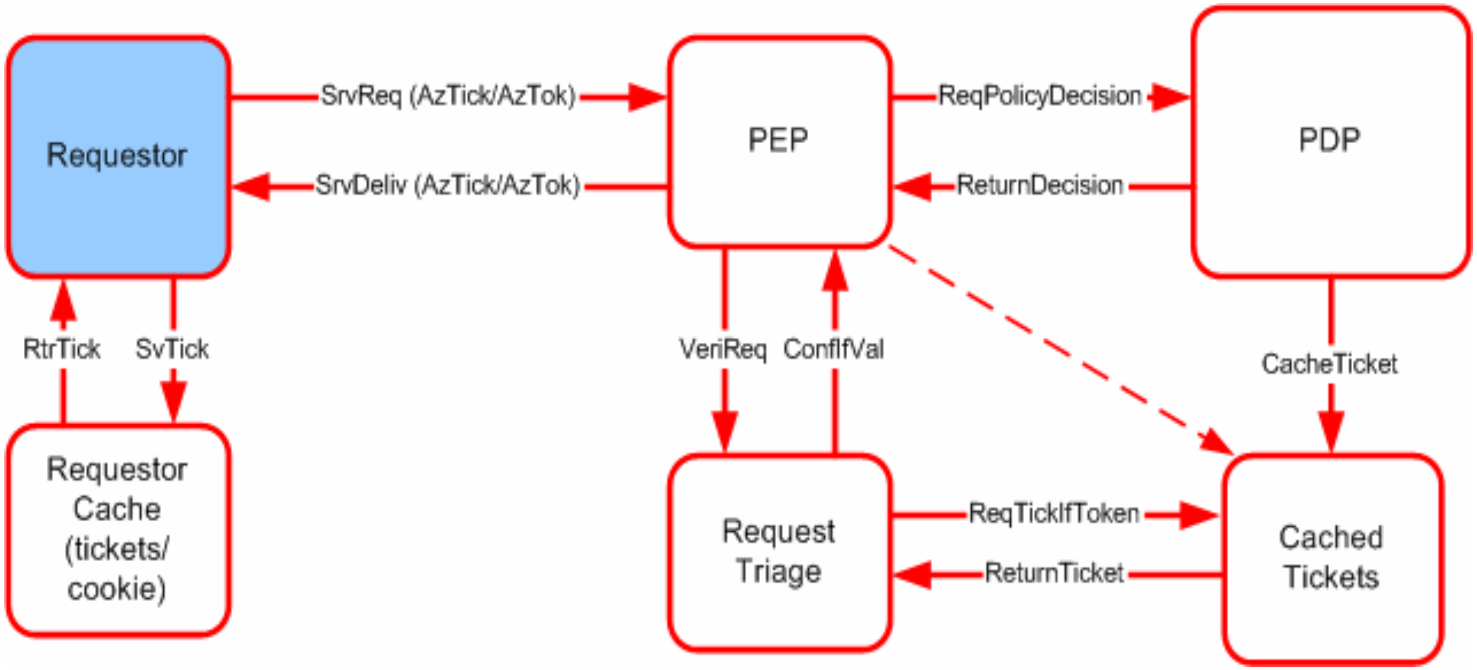
AuthzToken example – 293 bytes

- **<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

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

GAAAPI Components to handle AuthZ Tickets