The Globus Security Architecture

UK e-Science Core Programme Town Meeting
April 11, 2005, London, UK

Frank Siebenlist - Argonne National Laboratory
(franks@mcs.anl.gov)
http://www.globus.org/
Outline

- The Globus Toolkit (GT)
  - Grid Security Infrastructure (GSI)
- Standard and Buzzword Compliance
  - WSS, WS-I, SAML, XACML, GGF, OGSA, ...
- Policy, Policy, Policy....
  - Attributes
    - Shibboleth, SAML, X509-ACs, VOMS, etc.
  - Authorization
    - Call-out, SAML Authz, XACML, PC, PERMIS, AAA-tk, Delegation...
  - Audit
    - ...missing link...
- Layered Services
  - MyProxy, GridFTP, CAS, PURSE,...
- Big Picture & Futures
  - Apache, Naming, Renewable Refs, GridLogon, more Policy...
Globus Toolkit

- WS, WS-I & WSRF compliant toolkit
  - MLS & TLS support
- WSS, WS-I, X509 Identity/Attribute/Proxy-Certificate, (GGF-)SAML, XACML, PERMIS, VOMS compliant toolkit
- Different platform support
  - Java, C/C++, Python, .Net/C#
- (Security-)Integrated with higher-level Svcs
  - GridFtp, GRAM, MDS, MyProxy, PURSE, OGSA-DAI...
- Many, many parties involved
  - Customer-requirements driven
  - ... with commercial “versions”...
- Open Source
  - Apache-style license
Leverage (Open Source) Security Service Implementations

- **OpenSSL**
  - “native” Proxy Certificate support coming...
    (thanks to OpenSSL hacker Richard Levitte and KTH!)
- **Internet2’s OpenSAML**
  - Part of GT - used by CAS/GridShib/AuthzCallout/...
- **Internet2’s Shibboleth**
  - NSF funded GridShib project to “Grid-enable” Shibboleth
- **Sun’s open source XACML effort**
  - Integrate sophisticated policy decision engine in the GT
- **Futures: Permis, Handle System, XKMS, XrML, ...**
Security Services Objectives

- It’s all about “Policy”
  - (Virtual) Organization’s Security Policy
  - Security Services facilitate the enforcement

- Security Policy to facilitate “Business Objectives”
  - Related to higher level “agreement”

- Security Policy often delicate balance
  - More security ⇔ Higher costs
  - Less security ⇔ Higher exposure to loss
  - Risk versus Rewards
  - Legislation sometimes mandates minimum security
Agreement ⇔ VO Security Policy

(Business) Agreement

- Price
- Cost
- Obligations
- QoS
- T&Cs
- Security

Dynamic VO Security Policy

- Members
- Resources
- Roles
- Attribute mgmt
- Authz mgmt

Static Initial VO Security Policy

- Trust anchors
- (initial) members
- (initial) resources
- (initial) roles
- Access rules
- Privacy rules
OGSA Security Services

Requestor's Domain
- Attribute Service
- Trust Service
- Authorization Service
- Credential Validation Service
- Privacy Service
- Audit/Secure-Logging Service

Service Provider's Domain
- Authorization Service
- Trust Service
- Attribute Service
- Credential Validation Service
- Audit/Secure-Logging Service

WS-Stub
- Secure Conversation
- Bridge/Translation Service

Requestor Application
- Attribute Service
- Trust Service
- Authorization Service

Service Provider Application
- Attribute Service
- Trust Service
- Credential Validation Service

VO Domain
GT’s Attribute Assertion Support

- VOMS/Permis/X509/Shibboleth/SAML identity/attribute assertions
- Assertions can be pushed by client, pulled from a service, or are made locally available

GT-runtime has to mix and match all Attribute information a consistent manner, and present it to the subsequent Authz stage...
GT - Shibboleth Integration

- NSF-funded “GridShib” Project
  - http://grid.ncsa.uiuc.edu/GridShib/
- Leverage Shibboleth implementations and deployments
  - Sophisticated, policy controlled attribute service
  - Client-server interactions through WS-protocols
  - (optionally) preserve pseudonymity of client
- GridShib code will become part of GT
  - Transparent use of Shib servers in GT-runtime
  - For GT, Shib is “just an other” sophisticated Federation/Attribute Svc, like LDAP+ACs, SAML, PERMIS, VOMS
    - (Shib doesn’t do authz...(nor does it provide backend server))
- “Grid meets Shib” at 3:35pm
  - Von Welch(NCSA)
GT’s GGF’s Authorization Call-Out Support

- **GGF’s OGSA-Authz WG:**
  “Use of SAML for OGSA Authorization”
  - Authorization service specification
  - Extends SAML spec for use in WS-Grid
  - Recently standardized by GGF
- **Conformant call-out integrated in GT**
  - Transparently called through configuration
- **Permis interoperability**
  - Ready for GT4!
- **Futures...**
  - SAML2.0 compliance ... XACML2.0-SAML2.0 profile
XACML-SAML-2 Alternative

- XACML-2 Authz Query Interface better/superior/easier than (GGF) SAML-1 Authz equivalent
  - Tied integration with attributes
  - “obligations” part of the model
- XACML-2 Authz Query Message exchange is essentially “generic” and not tied to XACML
  - Other decision engines can be used behind implementation

- In GT & GGF, we’re “investigating” the use of the XACML’s request context and result as the common denominator...
Delegation Service

- Exposes delegated credentials as first class resource
- Allows for resource across multiple services
  - E.g. multiple jobs, RFT requests
- Allows for explicit destruction and renewal

- Brings delegation processing on the application level, such that PCs delegation certificate exchange can be supported by “all” toolkits
GT-XACML Integration

- eXtensible Access Control Markup Language (XACML)
  - OASIS standard
  - Open source implementations
- XACML: sophisticated policy language
- Globus Toolkit will ship with XACML runtime
  - Integrated in every client and server build on GT
  - Turned-on through configuration

- ...and we’re using the XACML-“model” for our Authz Processing Framework...

- ...can be called transparently from runtime and/or explicitly from application...
Propagation of Requester’s Rights through Job Scheduling and Submission Process

Virtualization complicates Least Privilege Delegation of Rights

Dynamically limit the Delegated Rights more as Job specifics become clear

Trust parties downstream to limit rights for you... or let them come back with job specifics such that you can limit them
GT’s Assertion Processing “Problem”

- VOMS/Permis/X509/Shibboleth/SAML/Kerberos identity/attribute assertions
- XACML/SAML/CAS/XCAP/Permis/ProxyCert/SPKI authorization assertions
- Assertions can be pushed by client, pulled from service, or locally available
- Policy decision engines can be local and/or remote
- Delegation of Rights is required “feature” implemented through many different means

GT-runtime has to mix and match all policy information and decisions in a consistent manner...
Attribute Collection Framework
GT’s Authorization Processing Model

- Use of a Policy Decision Point (PDP) abstraction that conceptually resembles the one defined for XACML.
  - Normalized request context and decision format
  - Modeled PDP as black box authorization decision oracle
- After validation, map all attribute assertions to XACML Request Context Attribute format
- Create mechanism-specific PDP instances for each authorization assertion and call-out service
- The end result is a set of PDP instances where the different mechanisms are abstracted behind the common PDP interface.
GT’s Authorization Processing Model (2)

- The Master-PDP orchestrates the querying of each applicable PDP instance for authorization decisions.
- Pre-defined combination rules determine how the different results from the PDP instances are to be combined to yield a single decision.
- The Master-PDP is to find delegation decision chains by asking the individual PDP instances whether the issuer has delegated administrative rights to other subjects.
- The Master-PDP can determine authorization decisions based on delegated rights without explicit support from the native policy language evaluators.
GT Authorization Framework (1)
GT Authorization Framework (2)

AAA/PERMIS/XACML

PDP

AAA

PDP

AAA

token

Request

saml authz assertion issued by AaId-2

xacml policy assertion issued by AaId-1

AAA

PDP

EPR

PDP

local PDP

resource owner (RsrId)

authzSvc config EPR + AaId-3

service Provider

stub-runtime

Policy Decision Point

xacml policy statement

EPR

AuthzSvc

local policy DB
GT Authorization Framework (3)
MyProxy/GridLogon

- No long-lived secrets on the user’s workstation
  - => move secrets to a secure MyProxy-server
    - Issue derived short-lived proxy-certificates
  - => issue short-lived identity certificates
    - On-line Certificate Authority (CA)
- Need for bootstrap authentication...
  - Passwords
  - One-Time-Passwords
- Need for “true” secure password protocol
- GridLogon would extend MyProxy
  - “simple” CA management
  - Trust-root provisioning of clients
OTP & Trust-Root Provisioning

Bootstrap User’s Trust-Root Config from Secure OTP Authentication

Secure mutual OTP-Authentication and Key-Exchange

Enhanced MyProxy/GridLogon Svc

OTP AuthN Server + user’s security config

Short-Lived Cert + Provisioning of CA’s, AuthZ/Attr Authorities

user-workstation (initially not configured)
Portal-based Grid Interface: PURSE

- Portal extensions (CGI scripts) that automate user registration requests.
  - Solicits basic data from user.
  - Generates cert request from CA (implemented with “simple CA” from GT).
  - Admin interface allows CA admin to accept/reject request.
  - Generates a certificate and stores in MyProxy service.
  - Gives user ID/password for MyProxy.

- Benefits
  - Users never have to deal with certificates.
  - Portal can get user cert from MyProxy when needed.
  - Database is populated with user data.

- This can be reused in other projects!
Earth Science Grid’s use of CAS-Assertions

MyProxy/GridLogon used for portal authentication

MyProxy/GridLogon used for UserDN mapping

Group membership assignment

Access Policy expressed with groups, actions and logical file names

Group | Operation | LFile

Mapping of logical file names to physical file paths

User with “UserDN” is allowed to invoke “Operation” on physical file “Pfile”
ESG External GridFTP Retrieval

GridFTP Server
- "CAS" policy enforcement

User
- gridftp access
  - GSI-creds
  - Portal authz assertion

MyProxy
- username
- userDN
- group
- Action
- LFile
- PFile

Portal
- policy enforcement
- login
- browse

PFile
- GridFTP Server
- PFile URL
  + authz assertion

Proxycert Issuance

Login

Proxycert Issuance
X.509 Proxy and End Entity Certificates still backbone of authentication and delegation
  - ...but support for more expressive assertion languages (SAML/XACML) will allow for alternatives...

Web Services technologies are providing more of the low-level plumbing
  - Use of SOAP-Header instead of ProxyCert embedding for communication of security info

Portals growing as a user interface
  - Clients use http, ... but portals will use WS-protocols!

New Deployment Paradigms (GridLogon, VMs)
  - Driven by our inability to protect the desktop...

Authorization still the big focus
  - “unification framework” needed to support different mechanisms and formats
GT - Futures

- Follow WSS, WS-I, OASIS, WSRF, GGF...
  - ...and solve strategic issues...
- GT-plumbing => Apache
  - ...long term strategy... (our concerns is higher up!)
- More Policy Integration
  - Security Policy Negotiation/Publishing/Discovery
  - Job Execution & Agreement Language Integration
    (?Semantic Web?)
  - Infrastructure Svc Integration to enable the “5-min VO”
  - GridLogon Provisioning
  - Secure Logging & Audit
  - Resource Reference Stability, resource migration, VMs
  - Extend use of Portals
  - Secure OTP
  - Kerberos
  - ... stay requirement driven - listen to our “customers”...