R-GMA Authorization Design

Linda Cornwall, RAL

24th February 2004

R-GMA and Authorization

- R-GMA is a distributed system with multiple onward connection whereby information is passed around different servlets
- There are 3 steps to ensure information is only available to authorized principals
- Firstly, ensure information is only passed to authorized principals
  - Essential, most of the rest of this presentation is about this
- Secondly, only put information into the system if requested by an authorized principal
  - No plans to do this at present – contradicts R-GMA design
- Thirdly, encrypt the information
  - Suspect will be necessary in the longer term

Authorizing the Service

- The various R-GMA servlets will need to be authorized to carry out the given role e.g.
  - To access information
  - To act as a Registry, Schema, Producer and Consumer
- Who decides authorization rules?
  - Both VOs and Resource controllers (or sysadmins) need to be able to set appropriate authorization rules on service Authorization
- R-GMA Service Authorization is an exercise in mutual Authorization

What the VO should specify

- Which hosts are trusted to take on the following R-GMA roles or capabilities:--
  - Registry
  - Schema
  - Producer
  - Consumer
- Which information suppliers (or Sensors) should be allowed to supply information to the VO
- Generic Authorization rules on data handled within the VO
  - Trusted hosts will enforce these rules.
  - ?? Should the trusted hosts be explicitly authorized to access this data or is it implied? (Implied would be simpler.)
- VOs may wish to specify more than 1 set of rules
  - E.g. they might trust a wide range of hosts to handle most information, but only allow limited access to highly confidential information. Maybe 2 capabilities, roles or Groups

Service Spec by VO

- VO should issue a VO signed specification of which services are authorized to handle information
- Authorization to carry out the R-GMA service roles may be based on:--
  - DN (or list of DNs)
  - VO membership
  - Group, Role, or Capability within the VO
- Normally, either a list of DNs, or an information role or Capability
- Allow the possibility for more than one specification within a VO – e.g. for different groups.
- If not based on a list of DNs, the VO will issue the appropriate credentials. (E.g. via VOMS)

Spec

- VO (and e.g. Group)
  - Registry – appropriate credentials
  - Schema – appropriate credentials
  - Producer – appropriate credentials
  - Consumer – appropriate credentials
  - Sensors – appropriate credentials
- Signed by the VO
- Principals accessing data will also need to know who to trust to supply info
  - (probably will need a copy of the same rules as the servlets)

VO service spec GAGL like e.g.


```
<gacl>
<entry>
<voms><fqan>/vo.dom.ain</fqan>
<capability>monreg</capability>
<allow><registry/><schema/><producer/><consumer/></allow>
</voms>
</entry>
<entry>
<voms><fqan>/vo.dom.ain</fqan>
<capability>mongen</capability>
<allow><producer><consumer></allow>
</entry>
<entry>
<voms><fqan>/vo.dom.ain</fqan>
<capability>monsensor</capability>
<allow><write></allow>
</entry>
```

VO credentials and VOMS

- If the specification is NOT based on a list of DNs, the VO will issue the appropriate credentials
- VOMS allows for a general `Capability' which is probably what is appropriate to allow for the various R-GMA roles for a given VO
- It would also allow for a flexible capability – so there could be more than 1 level of information service security within a VO
  - E.g. info service spec per group
- At present, no provision in VOMS to sign a spec for which credentials are needed within a VO

Resource or sysadmin Authorization

- The resource controller (or sysadmin) should be able to define who they allow to connect to their service
- We already have something in place – but this is not based on Grid credentials
  - We should move to something based on Grid credentials
- Not so important that it is signed (but could be to prevent hacking) as it is handled by local sysadmin

Authorization Enforcement (R-GMA service)

- Each Servlet will have to check that each other servlet is Authorized to act in the appropriate role for that VO
- The sensor will need to state which VO (and maybe e.g Group) for which it is producing information
- Each Producer will need to (for example)
  - Check the sensor writing is authorized by sysadmin
  - Check the sensor writing is authorized for the given VO (and possibly e.g. Group)
  - Check the Registry(s) Schemas(s) and Consumers it connects to are authorized by sysadmin
  - Check the Registry(s) Schema(s) and Consumer(s) are all Authorized to access data for that VO (and possibly e.g. Group)
- 2 layers of authorization enforcement – sysadmin and VO
- There may be more than 1 set of specs per VO.

Information Access rules

- Need to specify Authorization to access information by external principals.
- R-GMA is not as simple as `can this principal access this file', authorization needs to be based on views of the tables
- Need to develop a way of specifying how to carry out authz based on a view of a table
- MySQL doesn't at present support views – planned for the future
  - Didn't say exactly when

VO defined authorization rules

- VOs may define authorization rules on VO specific information
- These could be in the Schema
- Assuming we can have a Schema per VO
- Good for merging information from different sources
- Means it's not necessary to copy Authz rules with the info
- Good for allowing Access Control on any view you like
- `Mediator' can make a decision on what queries may be successful

VO Authz rules in Schema

- Take a fictional table having 5 columns (other than date and time) created with
- CREATE Table ConfJob (Jobid VARCHAR(200), State VARCHAR(30), Owner VARCHAR(200), Usage VARCHAR(50), JobDesc VARCHAR(200))
- Imagine the following:--
  - All info is available if the owner matches the DN
  - The JobDesc is only available to the Owner
  - The Usage is available to the VOadmin
  - Note that if you wanted a more specific rule, e.g. the full info is available to a supervisors DN which is row dependent – either need an additional table – or authz rules to go with the info

Schema for rules

- Add to the table a rule saying `full table not available to anyone'
  - Servlet access is assumed – as already authorized
- Add a rule saying `If DN=Owner – allow this row'
  - SELECT * from ConfJob where Owner=DN
  - This is a valid view statement – but only after name substitution?
- Add a rule saying `If Role=VOadmin – allow this table with JobDesc removed'
  - SELECT (Jobid, State, Owner, Usage) from ConfJob
  - This is a valid view statement.
- But where does MySQL allow such a spec?

`Per row' Authz

- CREATE Table ConfJob (Jobid VARCHAR(200), State VARCHAR(30), Owner VARCHAR(200), Usage VARCHAR(50), JobDesc VARCHAR(200), Authz VARCHAR(200))
- Authz on a row could be easily moved around.
- Would give `free' specification
- But difficult to do anything about a view – unless we encrypt the JobDesc.

edg-java-security

- edg-java-security authorization allows the appropriate credential access (DN, VO, Group, Role, and Capability)
- Not sure whether it will work (yet) for the client
- edg-java-security also implements a course grained authorization - which possibly could be used for authorization of the servlet roles –
  - but mutual authorization is not in place
  - 1 layer only
  - Unlikely to work