The VOMS System for Authorization Management inside Virtual Organizations

Vincenzo Ciaschini
INFN-CNAF
GGF School
Vico Equense, 22/7/2003

Outline

- Authorization in the Globus Toolkit
- Virtual Organization Membership Service
  - VOMS Server.
  - VOMS Client.
  - Administration Server.
  - Admin User Interface.
  - mkgridmap++
- VOMS in the EDG environment
- VOMS in VOX
- Future developments
- References

Part I:
Authorization in the Globus Toolkit

Security in the Globus Toolkit: Requirements

- Single sign-on.
  - The user should not be required to repeat login procedures on the grid more than once.
- Delegation.
  - Once a user has successfully identified himself with the Grid, it should be possible for grid services to act on the behalf of the user as if they were the user himself.
- User-based trust relationship.
  - All trust mechanism should have the user's credential at their core.
    - If a user wants to access farms A and B, there should be no need for farms A and B to trust each other.
- The user's credential should be adequately protected.
  - Private data (keys, passwords, etc…) should not circulate on the net.

Security in the Globus Toolkit: Requirements (final)

- Integrated with local systems.
  - The grid security mechanism should not supplant the local authorization mechanism, but instead work on top of it.
- Simple to use.
  - The system should be simple enough on the user's side as not to require excessive preparations before real work could begin.
- The system used should employ well defined standards to permit multiple implementations.

Security in the Globus Toolkit: The Solution

- Protocols: X.509 certificates, PKI, GSS-API and GSI.
- X.509 certificates:
  - An ISO and IETF standard that ties public key credentials (public and private keys) to an identity.
  - Certificates are issued by a set of well-defined Certification Authorities (CAs).
  - Credentials are divided in two parts:
    - The public part in the certificate, supposed to be shared.
    - The private part, that must be kept secret by the user.

Security in the Globus Toolkit: The Solution (cont'd)

- PKI:
  - Public Key Infrastructure.
  - A set of IETF standards that define how the certificates and CAs must work together.
- GSS-API:
  - Generic Security Services Application Program Interface.
  - An IETF standard that defines a unified interface to heterogeneous security mechanisms (Kerberos, X.509 certificates, etc…).

Security in the Globus Toolkit: The Solution (final)

- GSI:
  - Globus Security Infrastructure.
  - Ties together the other three components.
  - Adds the capabilities of credentials delegation.
  - Defined in a set of documents on the Globus site (http://www.globus.org)

Sample Certificate

Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number: 1148 (0x47c)
    Signature Algorithm: md5WithRSAEncryption
    Issuer: C=IT, O=INFN, CN=INFN Certification Authority
    Validity
      Not Before: Jan 31 13:29:07 2003 GMT
      Not After : Jan 31 13:29:07 2004 GMT
    Subject: C=IT, O=INFN, OU=Personal Certificate, L=CNAF, CN=Vincenzo Ciaschini/Email=Vincenzo.Ciaschini@cnaf.infn.it
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public Key: (1024 bit)
        Modulus (1024 bit):
            …..
        Exponent: 65537 (0x10001)
    X509v3 extensions:
      X509v3 Basic Constraints: critical
        CA:FALSE
      X509v3 Key Usage: critical
        Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment
Signature Algorithm: md5WithRSAEncryption
Signature: …

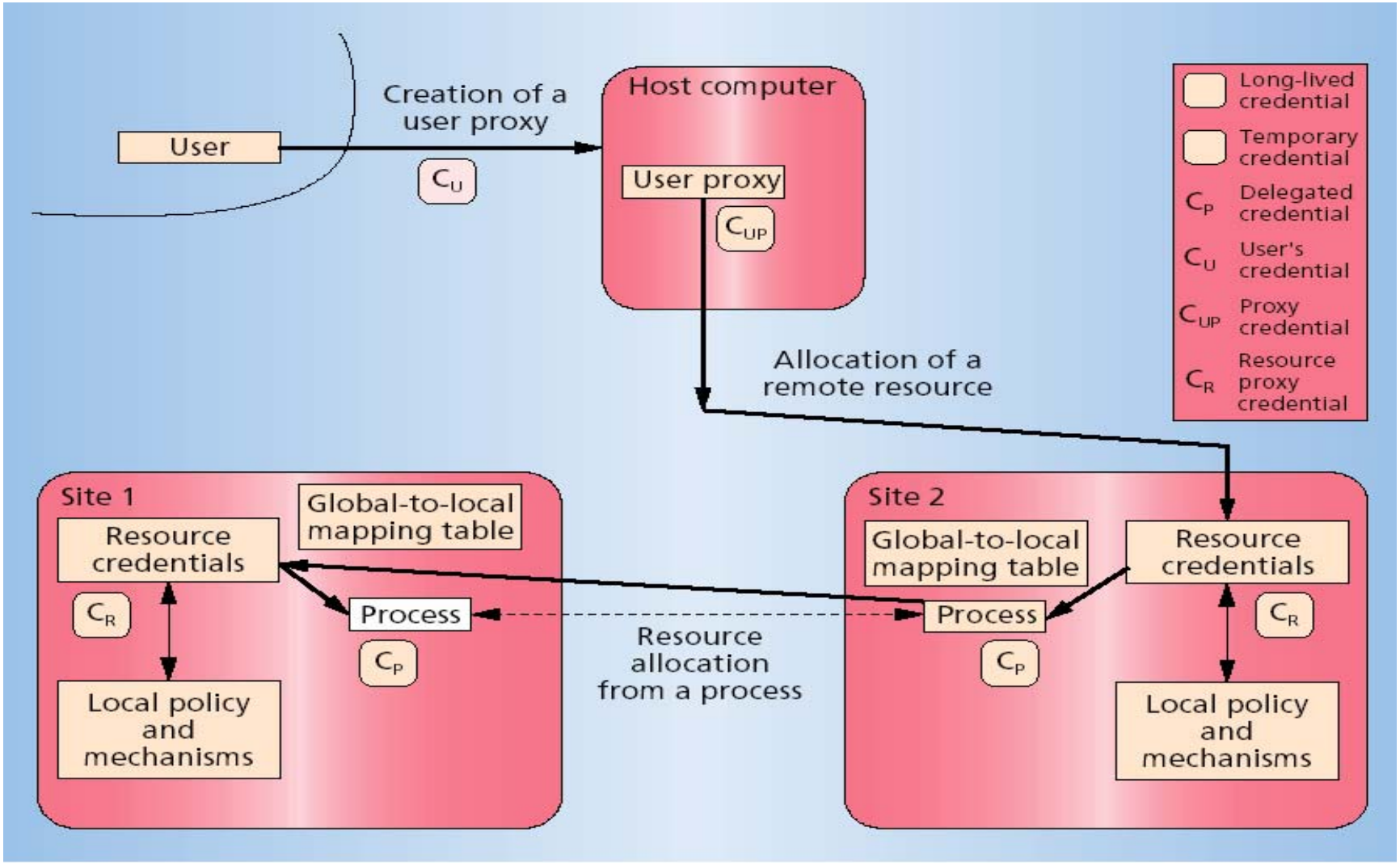
Sample certificate (real data)

-----BEGIN CERTIFICATE-----
MIIFXzCCBEegAwIBAgICBHwwDQYJKoZIhvcNAQEEBQAwQzELMAkGA1UEBhMCSVQx
DTALBgNVBAoTBElORk4xJTAjBgNVBAMTHElORk4gQ2VydGlmaWNhdGlvbiBBdXRo
b3JpdHkwHhcNMDMwMTMxMTMyOTA3WhcNMDQwMTMxMTMyOTA3WjCBlzELMAkGA1UE
BhMCSVQxDTALBgNVBAoTBElORk4xHTAbBgNVBAsTFFBlcnNvbmFsIENlcnRpZmlj
YXRIMQ0wCwYDVQQHEwRDTkFGMRswGQYDVQQDExJWaW5jZW56byBDaWFzY2hpbmkx
LjAsBgkqhkiG9w0BCQEWH1ZpbmNlbnpvLkNpYXNjaGluaUBjbmFmLmluZm4uaXQw
gZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAM6xlVewokq1+2HgBGdVE3t51Kv4
hiCEFd5uXzwpUM+Z6dkBHucSO6m28PnRGdFOb8tfpY/+Ysku/BCAYLVfbEhDuat6
0DCDRzMM1i+IWUJJ5EgBa7CWdkuJPabf6/aiHbWgqctTo6V3NwN2ouAHOSBJjrzl
3D27svZpbBcl3yGXAgMBAAGjggKKMIIChjAMBgNVHRMBAf8EAjAAMA4GA1UdDwEB
/wQEAwIE8DA2BgNVHR8ELzAtMCugKaAnhiVodHRwOi8vc2VjdXJpdHkuZmkuaW5m
bi5pdC9DQS9jcmwuY3JsMBcGA1UdIAQQMA4wDAYKKwYBBAGIEwoBATAdBgNVHQ4E
FgQUQ5IXNTUVcaiBjwTDFojCdYQ6Sk4wawYDVR0jBGQwYoAUyhHvXR0HBJippbVY
GmZOChYr4EmhR6RFMEMxCzAJBgNVBAYTAklUMQ0wCwYDVQQKEwRJTkZOMSUwIwYD
VQQDExxJTkZOIENlcnRpZmljYXRpb24gQXV0aG9yaXR5ggEAMCoGA1UdEQQjMCGB
H1ZpbmNlbnpvLkNpYXNjaGluaUBjbmFmLmluZm4uaXQwPQYDVR0SBDYwNIESaW5m
bi1jYUBmaS5pbmZuLml0hh5odHRwOi8vc2VjdXJpdHkuZmkuaW5mbi5pdC9DQS8w
EQYJYIZIAYb4QgEBBAQDAgWgMFcGCWCGSAGG+EIBDQRKFkhJc3N1ZWQgdW5kZXIg
SU5GTiBDQSBDUCBhbmQgQ1BTIHYwLjMsIGh0dHA6Ly9zZWN1cml0eS5maS5pbmZu
Lml0L0NBL0NQUy8wKgYJYIZIAYb4QgECBB0WG2h0dHA6Ly9zZWN1cml0eS5maS5p
bmZuLml0LzAkBglghkgBhvhCAQMEFxYVY2dpLWJpbi9jaGVjay1yZXYucGw/MCYG
CWCGSAGG+EIBBwQZFhdjZ2ktYmluL2NoZWNrLXJlbmV3LnBsPzA4BglghkgBhvhC
AQgEKxYpaHR0cDovL3NlY3VyaXR5LmZpLmluZm4uaXQvQ0EvcG9saWN5Lmh0bWww
DQYJKoZIhvcNAQEEBQADggEBAFxJlznIQqvPSkaAAK2/luUh2ECOEXiLyFCzS7Ry
200+KnsgQZhXTlDTlFaGXGiK4Y6mDu3bkQiFCKRkVw/6EbFEtFRyjHddbDlfc0My
Cj4C5AKZzRWYHVO/MliQwOQh7jYqBM/tdKPbPTHKECyX1+o7BYLUdd1EI1OXgLG6
Tccw61KeFzLKZA5g45X+WGFxRvIrNtS1NkOxhWFNslFZRdGu9DGrbLap9QU19+oN
ZQwBSiK2G2yxQZEXddP/yJpgLHQAXsPLSrTqAXfG+RnRuCaWT6zjCPLK6wMaQ0y0
HDgcP3Y7i04RX+KNSDMJ5160iGSawRNWqJRDV9Krv1gTVWY=
-----END CERTIFICATE-----

Delegation

- Essentially create a new short-lived certificate (proxy) based on the existing one.
  - Done by the grid-proxy-init command.
- The original certificate never travels through the net, thus remaining secure.
- On the contrary, the proxy certificate travels on the net, but due to the short life, potential damages are restricted.

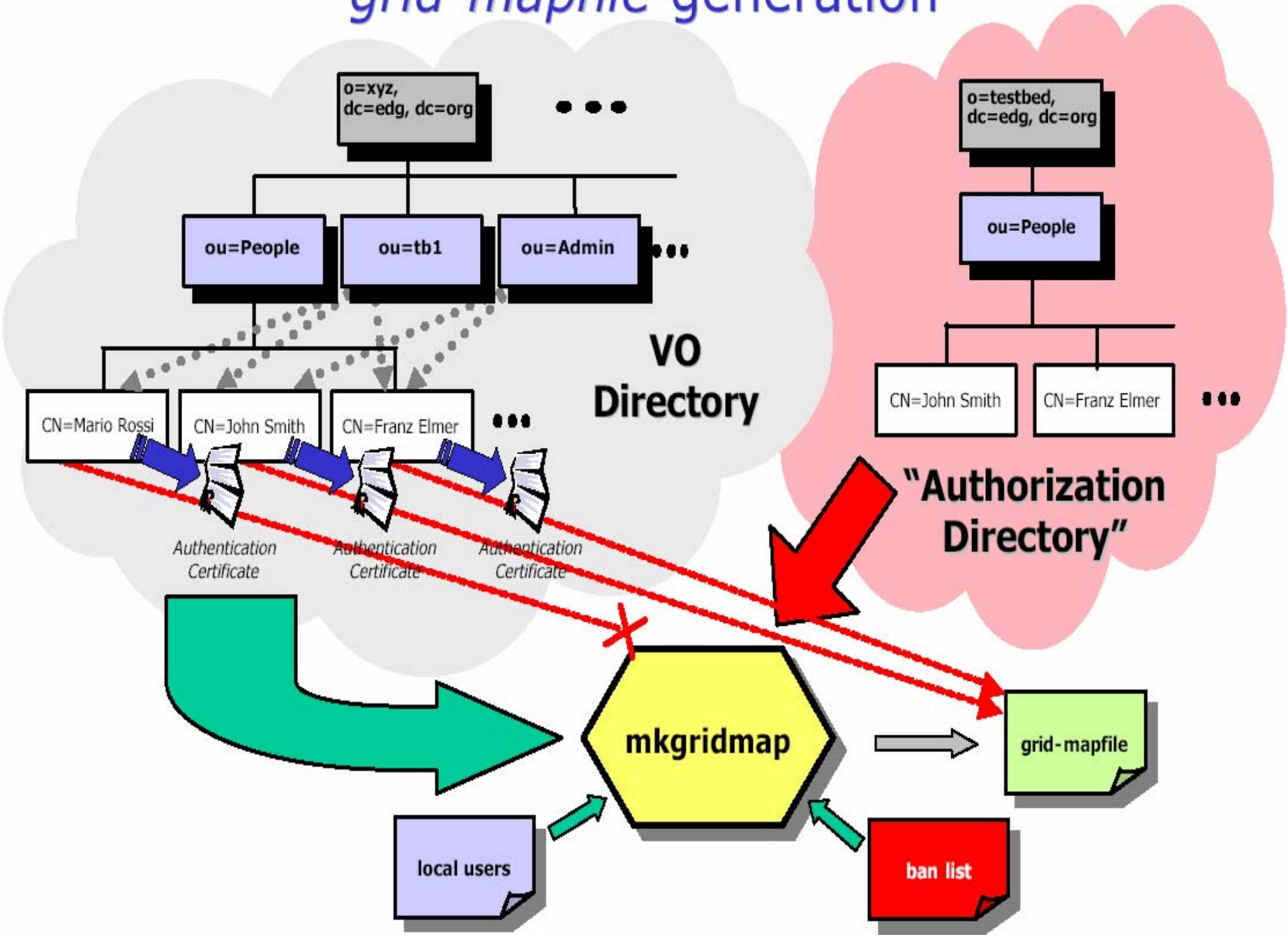
Authentication process: User side



Authentication Process: Farm side

- Based on matching on a list of accepted users (grid-mapfile).
- Maps remote credentials into local users.
  - May be done in a semi-dynamic way (see later).

grid-mapfile generation



Sample mkgridmap.conf

```
#### GROUP: group URI [lcluser]
group ldap://grid-vo.nikhef.nl/ou=tb1,o=atlas,dc=edg,dc=org
group ldap://grid-vo.nikhef.nl/ou=tb1,o=cms,dc=edg,dc=org .cms
group ldap://grid-vo.cnaf.infn.it/ou=tb1,o=cdf,dc=edg,c=it
#### DEFAULT LOCAL USER: default_lcluser lcluser|AUTO
default_lcluser AUTO
#### AUTHORIZED VO: auth URI
auth ldap://marianne.in2p3.fr/ou=people,o=tb,dc=edg,dc=org
#### ACL: deny|allow pattern_to_match
allow *INFN*
#### GRID-MAPFILE-LOCAL
gmf_local /opt/edg/etc/grid-mapfile-local
```

Problems:

- Very coarse-grained authorization:
  - Remote users are mapped directly to UNIX users.
  - Classification of users into categories must be done on a local farm basis without input from the VO (may result in the same user having very different privileges in different farms).
  - No support for groups or roles
  - Grid-mapfile authorization is not flexible.

Part II:
Virtual Organization Membership Service

Virtual Organization Membership Service

- VOMS for short.
- Developed for DataTAG by INFN (core services) and DataGrid by CERN (admin interface).

VOMS Objectives and requirements

- To provide a secure system for Virtual Organizations (VOs) to organize users into groups and/or roles and to disseminate this information.
  - A VO is a collection of users and resources working together on a common project.
  - Membership in a VO is a restricted information.
- Extensibility.
- Users should be able to specify how much information they want to publish.
- Backwards compatibility with the Globus Toolkit.
- Should not invalidate established GT-based work mechanisms.
- Should minimize software requirements other than GSI libraries in the core components.

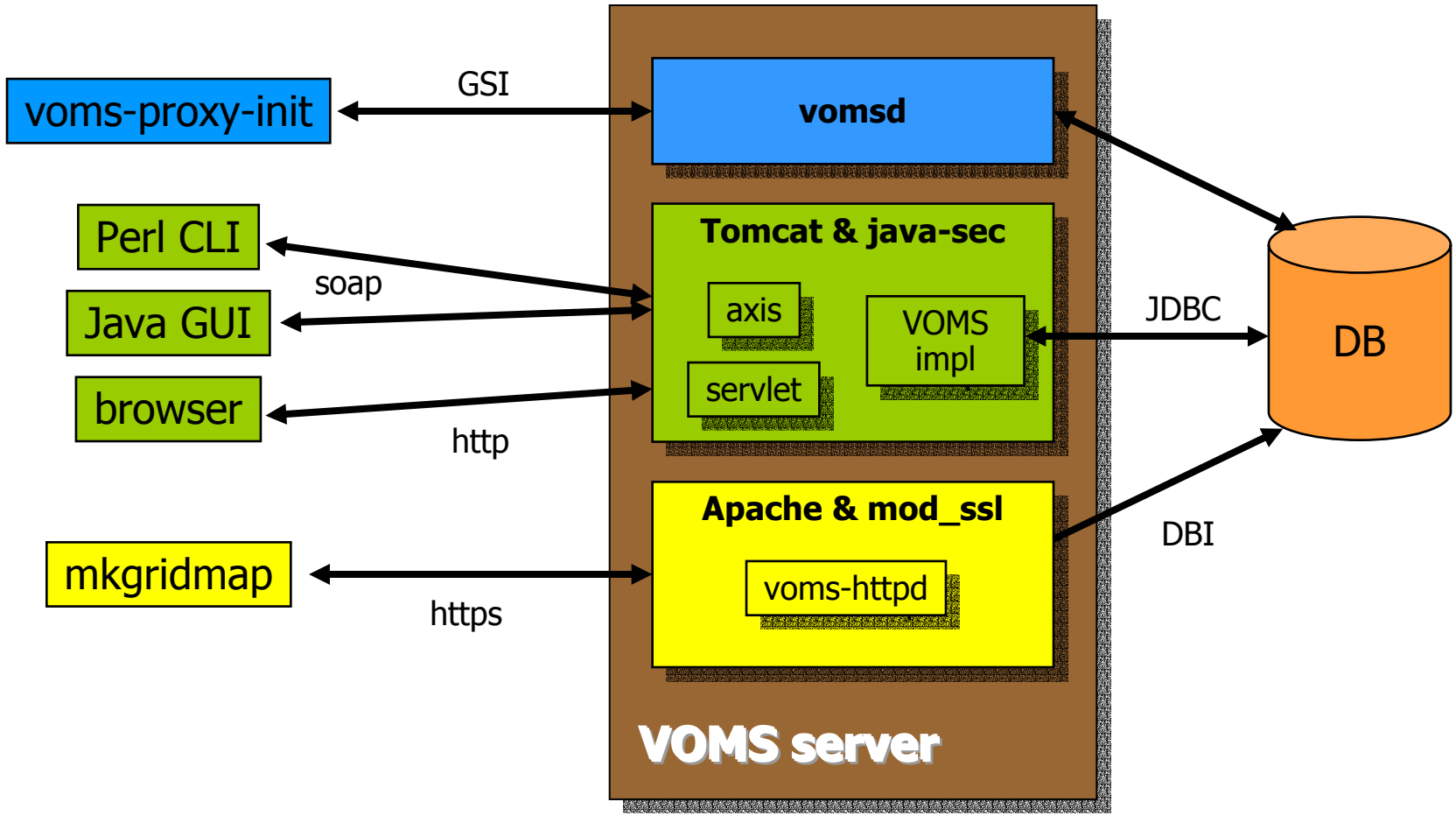
VOMS Solution

- Grant authorization at the VO level.
  - Each VO has its own VOMS server.
  - Contains (group / role / capabilities) triples for each member of the VO.
  - Also support for “forced groups” (for negative permissions.)
- Insert these information in a well-defined non critical extension of the user proxy.
- All client-server communication is secure and authenticated.
- Authorization info must be processed by the local sites.

VOMS Solution (cont'd)

- Based on RDBMS.
- Five primary components:
  - User client − queries the server for authorization info
  - Core server − returns authorization info to the client
  - Administration client − used by VO administrators for management
  - Administration server − executes client update operations on db
- Transition tool − interface to mkgridmap++ (see below)
- APIs
  - C and C++ APIs to access the extensions managed by VOMS and to let a program contact the server.

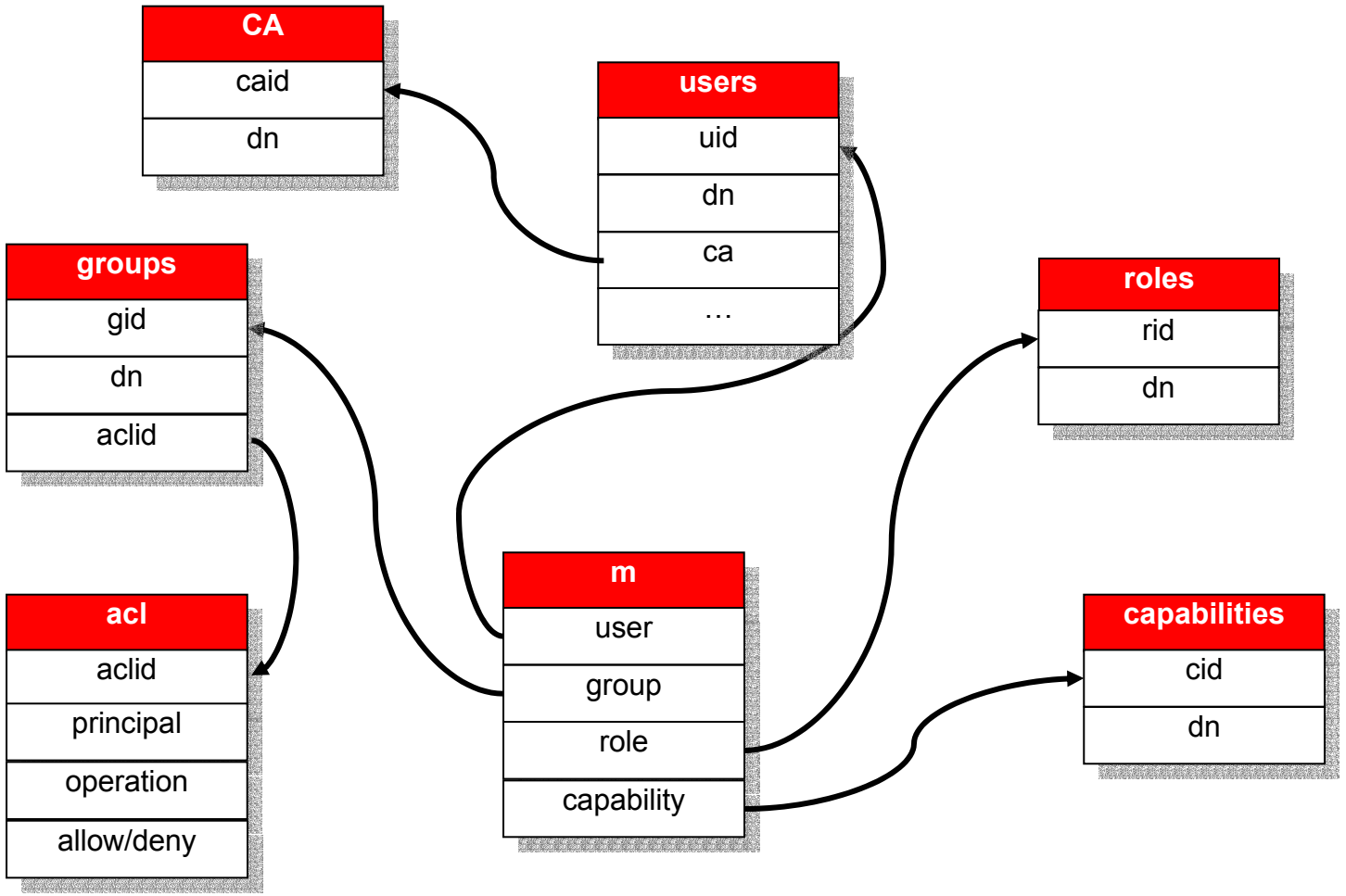
VOMS Architecture



VOMS Server

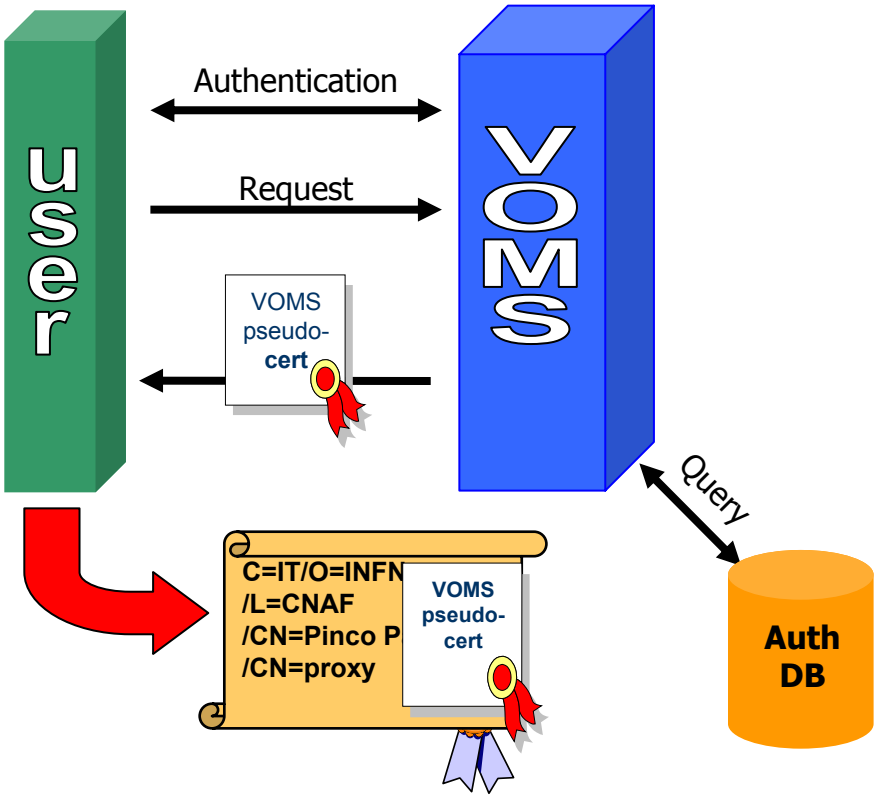
vomsd

Data Base Structure



vomsd

VOMS server operating scheme

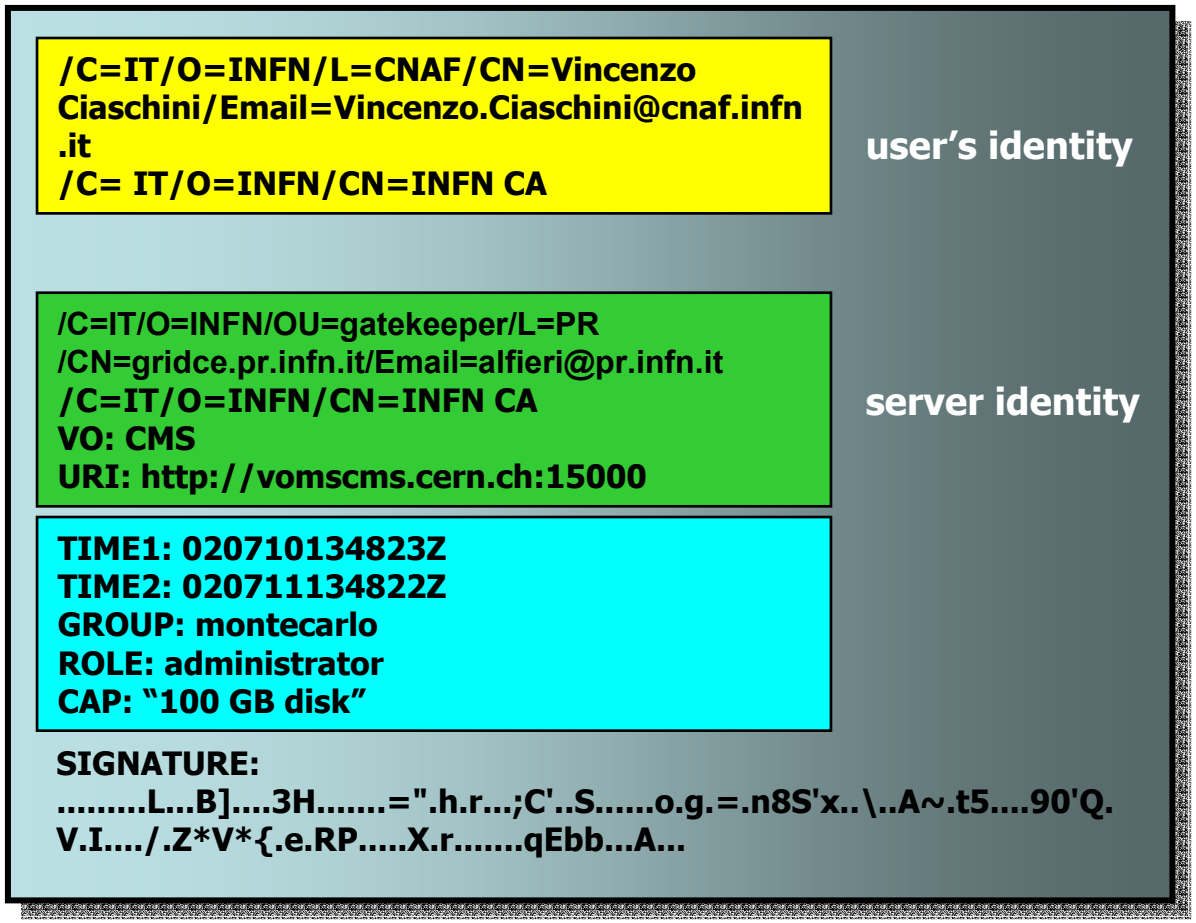


1) Mutual authentication between client and server.
   - Secure communication channel via Globus GSI.
2) The client sends request to server.
3) The server checks correctness of request.
4) The server sends back the required info in a “pseudo certificate” signed by itself.
5) The client checks the consistency and validity of the informations returned.
6) Steps 1-6 may be repeated for any number of servers.
7) The client creates a proxy certificate that includes the informations returned by the VOMS servers into a non critical extension.
8) Finally, the client may opt to include also additional information provided by the user.

vomsd

Pseudo Certificate Format

- This Pseudo Certificate is included into a non critical extension of the user's proxy .
  - OID:1.3.6.1.4.1.8005.100.100.1
- Conversion to a true attribute certificate already started.
- There will be one such certificate for each VOMS server contacted.



vomsd

Server software requirements

- GSI version 2.0 or higher.
- Database chosen: MySQL >= 4.0.13
  - Easily portable to other databases (PostgreSQL, Oracle). DB Access code is neatly separated from the rest and the DB schema should be portable.
- No other external software needed.

vomsd

VOMS Client

voms-proxy-init

edg-voms-proxy-init

- Drop down replacement for grid-proxy-init.
- Adds the ability to contact multiple VOMS servers and milk them for information.
- All connections made require mutual authentication, confidentiality and integrity.
- Also known as voms-proxy-init for compatibility with previous versions of VOMS.

voms-proxy-init

edg-voms-proxy-init invocation

- All the options accepted also by grid-proxy-init.
- Among the others:
  - **--voms** <server[:command]>
    - Contacts <*server*> for information, sending it <*command*>. May be:
      - **A** send all known informations. The defaut if <*:command*> is not specified.
      - **G<id>** send only informations related to group **id** and its ascendants.
      - **R<id>** send only informations relating to role **id** and to ascendants.
      - **B<id1>:<id2>** combine **G** and **R** commands, working on group **id1** and role **id2**.
      - **L** list all extended commands.
      - **S<num>** executes extension command **<num>**.
    - Almost all other options become meaningless if this is absent.
    - THERE IS NO DEFAULT SERVER.
    - More then one such option may appear. They will be processed in order.

edg-voms-proxy-init invocation (final)

- **--print** prints the informations returned by the servers on screen instead that generating the proxy.
- **--noregen** avoids generating an initial proxy for connection to the servers. Useful in conjunction the KCA.
- **--vomslife <num>** specifies a maximum validity (in hours) for the validity of the VOMS informations.
  - May only reduce the validity that the server would set.
  - The default is as long as the including proxy certificate.
- **--include <file>** includes a user specified file in the user's proxy. May contain additional authentication info, e.g. Kerberos ticket.
- **--order <group[:role]>** groups and roles will be returned by the server in the same order as specified by these options.
  - Multiple copies of these options may appear. They will be processed in order.
  - The default order is unspecified.

edg-voms-proxy-init setup

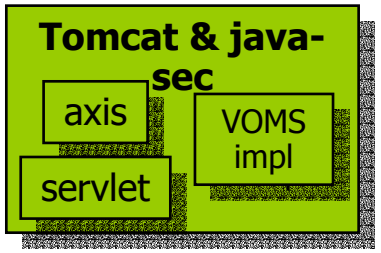
- Mutual authentication requires the subject of the server's certificate to be known beforehand.
- Along with the other needed data (hostname, port) for each server, would make the commandline unwieldy.
- Solution: Define a system that would associate to each server a nickname and use such nickname on the command line.
  - The association nickname-server data is done in a configuration file (/opt/edg/etc/vomses by default). This is the only configuration that should be done on the client machine.

Client software requirements

- GSI version 2.0 or higher.
- No other software required. ☺

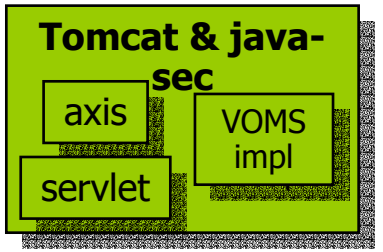
voms-proxy-init

Administration server



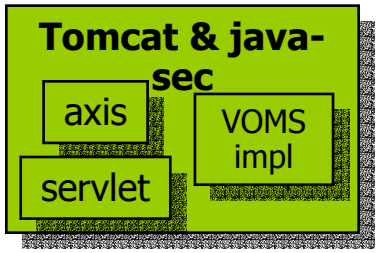
Admin server features

- May create a hierarchy of administrators, each with rights on subset of the VO structure.
- Administrators are identified with certificates.
- Keeps an history of all the changes done to the data.
- Administrator capabilities are defined in a set of ACLs.
- Administrators may control the ACLs of lesser administrator.
- It is always possible to directly access the DB in case of major goof-ups.



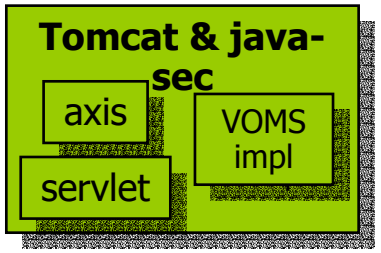
Administrator server capabilities

- May add/remove users / groups / roles / capabilities.
- May assign/remove users to groups/roles.
- May assign/remove capabilities to users/groups/roles.



Software requirements

- Java 1.4.x, Tomcat 4, edg-java-security, …
- Globus GSI 2.0 or higher.



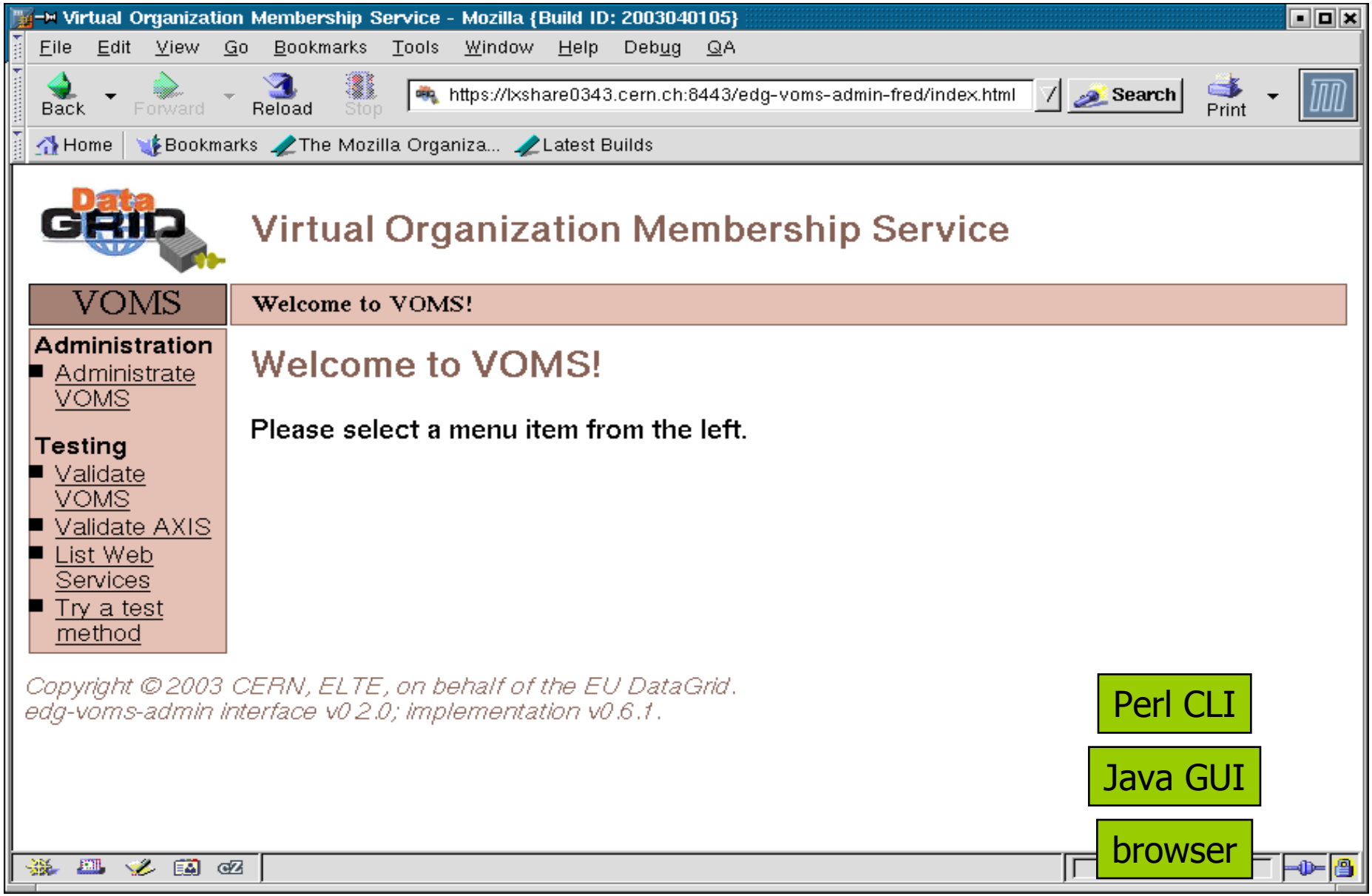
Admin interface client

Perl CLI

Java GUI

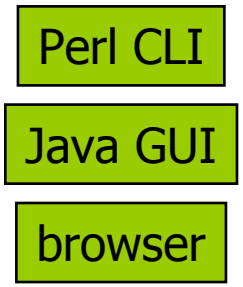
browser

Admin interface screenshot

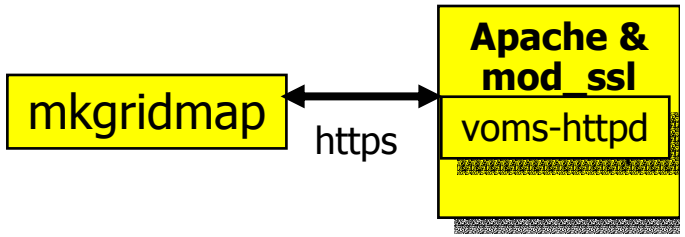


Software requirements

- Depend on the particular interface used:
  - Browser interface.
    - A browser with your own certificate installed.
  - Perl CLI interface.
    - Perl 5 and some modules (Soap interface).
  - Java interface.
    - Java 1.4.x and some classes (Soap interface).

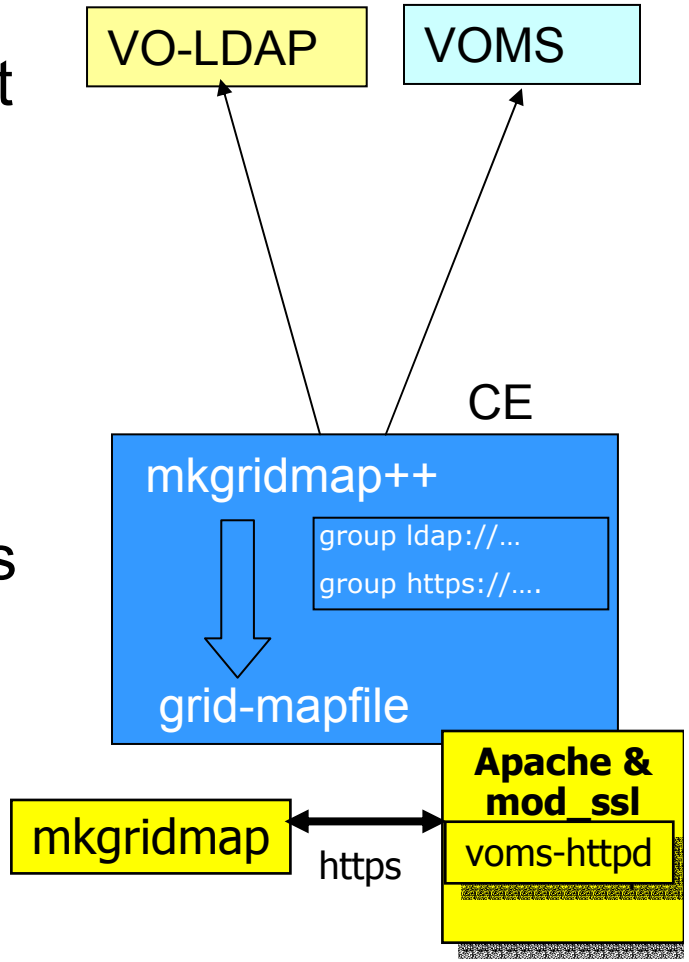


mkgridmap++



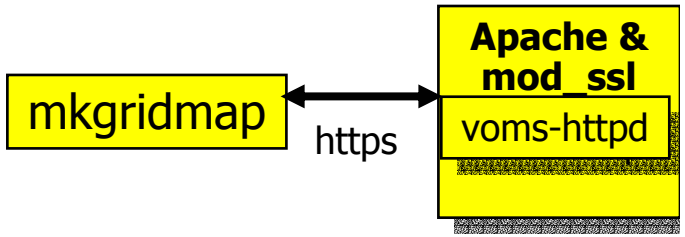
mkgridmap++ workings

- Able to contact indifferently LDAP or VOMS VOs.
  - VOMS and LDAP VOs can coexist pacifically in the same grid.
  - Uses a new directive completely similar to the one already existing.
- New feature:
  - Authenticated access to VOMS (*not LDAP*) servers based on https protocol to restrict the clients allowed to download the list of the VO members



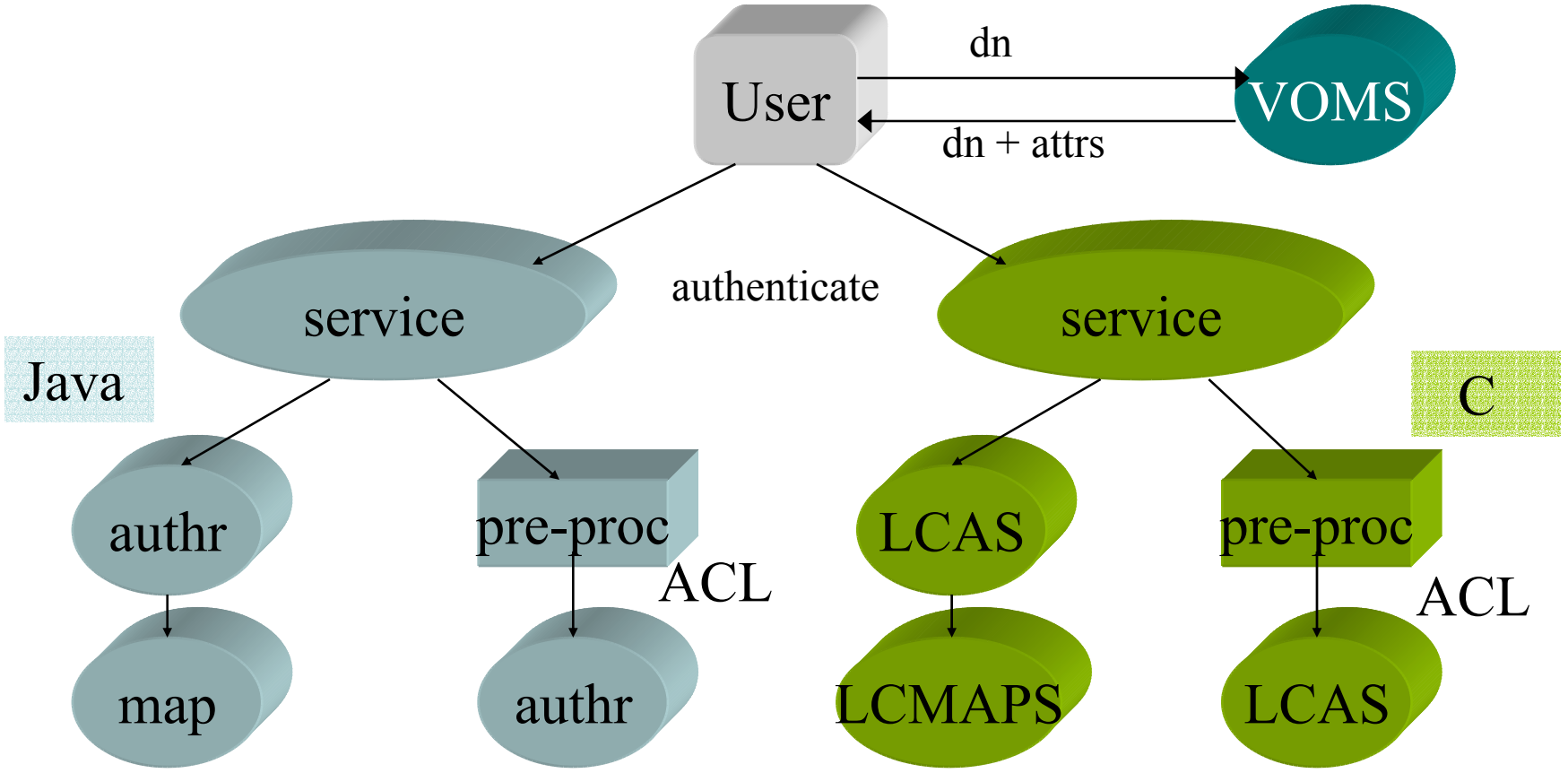
Software requirements

- Perl 5 and a whole lot of perl modules. (The same as plain mkgridmap plus a few more)



Part III
VOMS in the EDG environment

Authorization



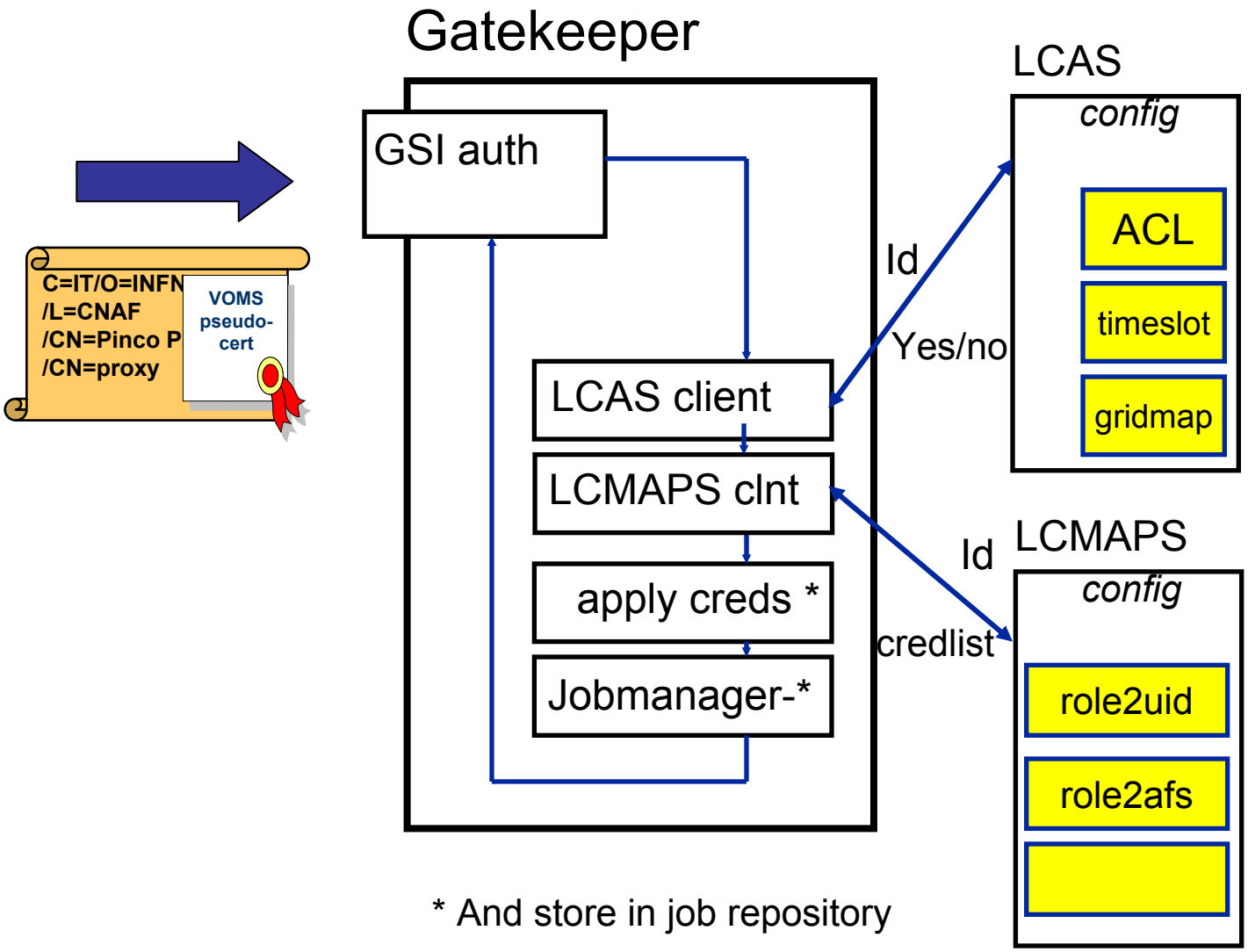
Coarse-grained
e.g. Spitfire

Fine-grained
e.g. RepMeC

Coarse-grained
e.g. CE, Gatekeeper

Fine-grained
e.g. SE, /grid

Fabric Access – EDG Style



Local Site Authorization Services

- Local Centre Authorization Service (LCAS)
  - Handles authorization requests to local fabric:
    - Authorization deision based on proxy and job specification.
    - Supports grid-mapfile mechanism.
  - Plug-in framework.
    - Allowed users.
    - Banned users.
    - Available timeslots.
    - Plug-in for VOMS.
- Local Credential Mapping Service (LCMAPS)
  - Provides local credentials neede for jobs in fabric.
  - Mapping based on user identity, VO affiliation, site-local policy.
  - Supports standard UNIX credentials and pool accounts
  - Plug-in framework
    - Plug-in for VOMS

Java Side

- Java Trustmanager:
  - Certificate validator for Java services.
  - Permits (mutual) secure authentication.
  - Uses standard X.509 certificates.
  - Supports authorization decisions using VOMS extensions.

Pout-pourri

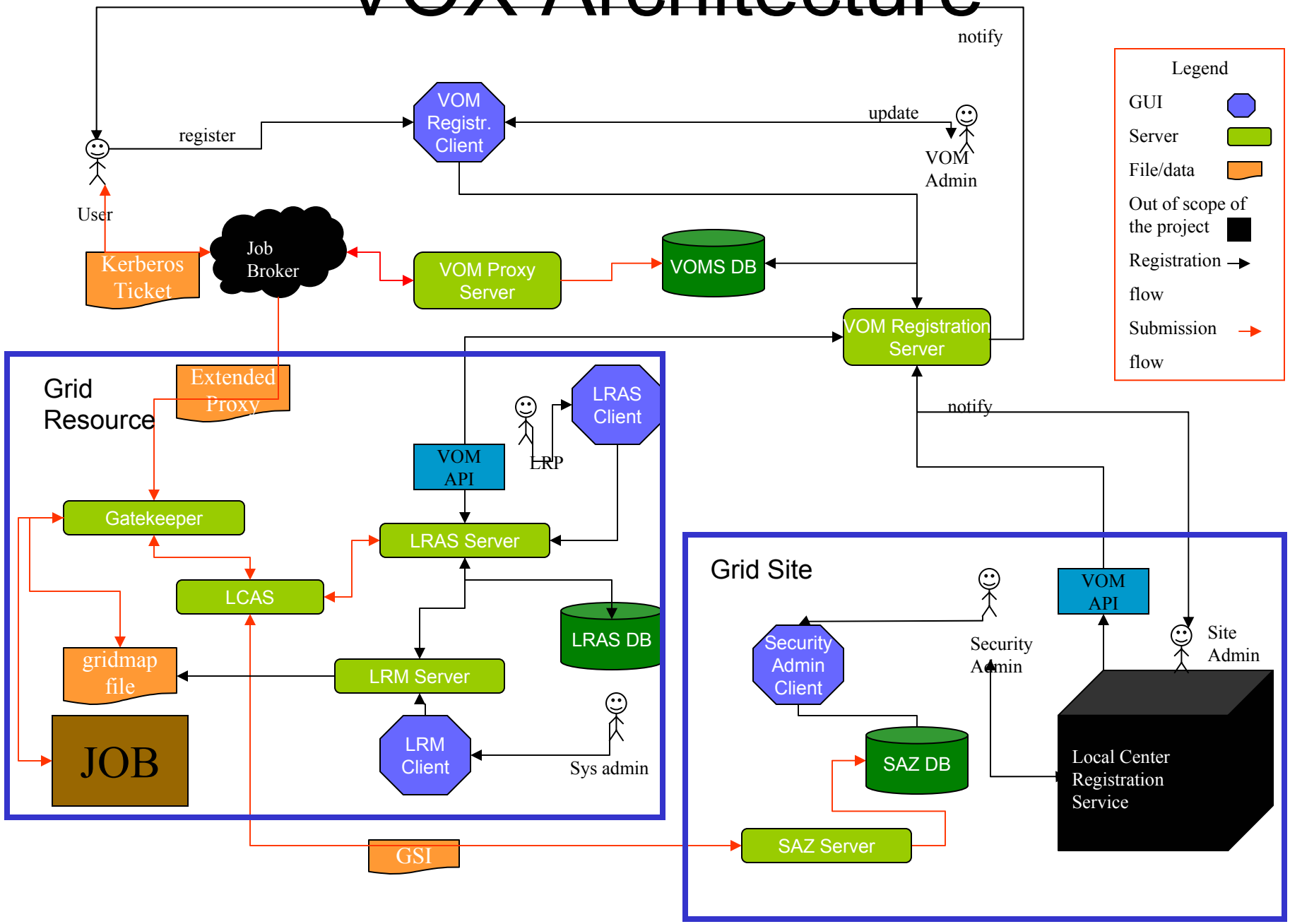
- Logging and Bookkeeping
  - Uses VOMS credentials to authorize access to data.
- R-GMA
  - Uses VOMS to authorize access to data. (in the future)
- Medium-grained authorization
  - Uses VOMS groups/roles to define mapping in intermediate accounts.

Part IV
VOMS in VOX

VOMS eXtension

- Collaboration between CMS US and DataTAG.
- Plans to develop and implement a complete user registration infrastructure around VOMS.
- Also plans to implement a new local authorization schema.

VOX Architecture



Part V
Future Developments

Future developments in VOMS

- Already started:
  - Replica system for the VOMS server.
  - Creation of true Attribute Certificates instead of Pseudo Certificates.
- Other developments:
  - Complete implementation of temporal checks on groups/roles.
  - Better logging.

Part VI
References

IETF References

- X.509 Certificates and PKI
  - RFCs 2459, 2510, 2511, 2527, 2528, 229, 2560, etc…
- GSS-API
  - RFCs 2078, 1964, 2744, 2853, etc…

- All this and others my be found on the IETF site at http://www.ietf.org

Globus References:

- **The Anatomy of the Grid: Enabling Scalable Virtual Organizations**. I. Foster, C. Kesselman, S. Tuecke. *International J. Supercomputer Applications*, 15(3), 2001.
- **A Security Architecture for Computational Grids**. I. Foster, C. Kesselman, G. Tsudik, S. Tuecke. *Proc. 5th ACM Conference on Computer and Communications Security Conference*, pp. 83-92, 1998.
- **A National-Scale Authentication Infrastructure**. R. Butler, D. Engert, I. Foster, C. Kesselman, S. Tuecke, J. Volmer, V. Welch. *IEEE Computer*, 33(12):60-66, 2000.

- All these can be found on the Globus site at http://www.globus.org

VOMS References:

- **VOMS: an Authorization System for Virtual Organizations.** Alfieri, Cecchini, Ciaschini, Dell'Agnello, Frohner, Gianoli, Lörentey, Spataro, *1st European Across Grids Conference, Santiago de Compostela*, February 13-14, 2003.
- **Managing dynamic user communities in a Grid of autonomous resources.** AAVV, *Chep 2003*
- **The VOMS CVS**. http://cvs.infn.it

- All this may be found on the Authorization Group DataTAG website: http://grid-auth.infn.it