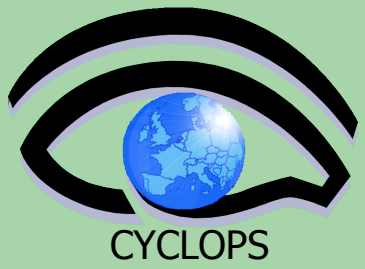
Authorization and Authentication in gLite



Emidio Giorgio
INFN Catania

Tutorial CYCLOPS

Bologna 11 April, 2007

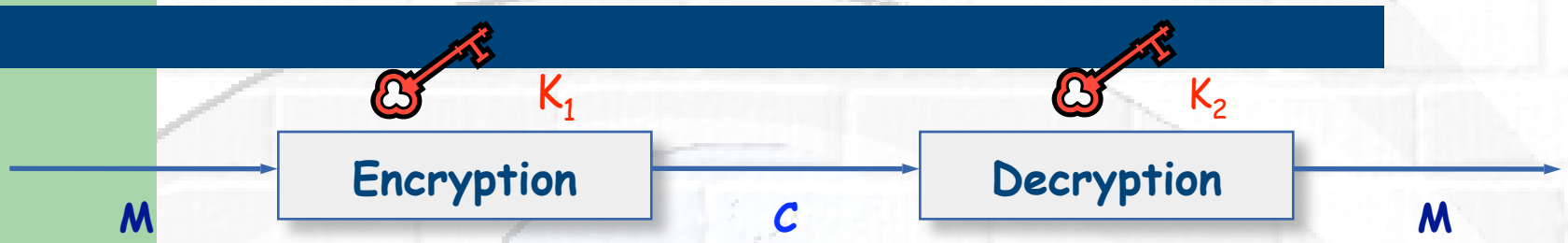
Overview

- Glossary
- Encryption
  - Symmetric algorithms
  - Asymmetric algorithms: PKI
- Certificates
  - Digital Signatures
  - X509 certificates
- Grid Security
  - Basic concepts
  - Grid Security Infrastructure
  - Proxy certificates
  - Command line interfaces
- Virtual Organization
  - Concept of VO and authorization
  - VOMS, LCAS, LCMAPS

Glossary

- **Principal**
  - An entity: a user, a program, or a machine
- **Credentials**
  - Some data providing a proof of identity
- **Authentication**
  - Verify the identity of a principal
- **Authorization**
  - Map an entity (principal) to some set of privileges
- **Confidentiality**
  - Encrypt the message so that only the recipient can understand it
- **Integrity**
  - Ensure that the message has not been altered in the transmission
- **Non-repudiation**
  - Impossibility of denying the authenticity of a digital signature

Symmetric/Asymmetric cryptography



Mathematical algorithms providing important building blocks for the implementation of a security infrastructure

- **Symbology**

  given a plaintext ***M*** and a cyphered text ***C***

  - Encryption with key K_1 : $E_{K_1}(M) = C$
  - Decryption with key K_2 : $D_{K_2}(C) = M$

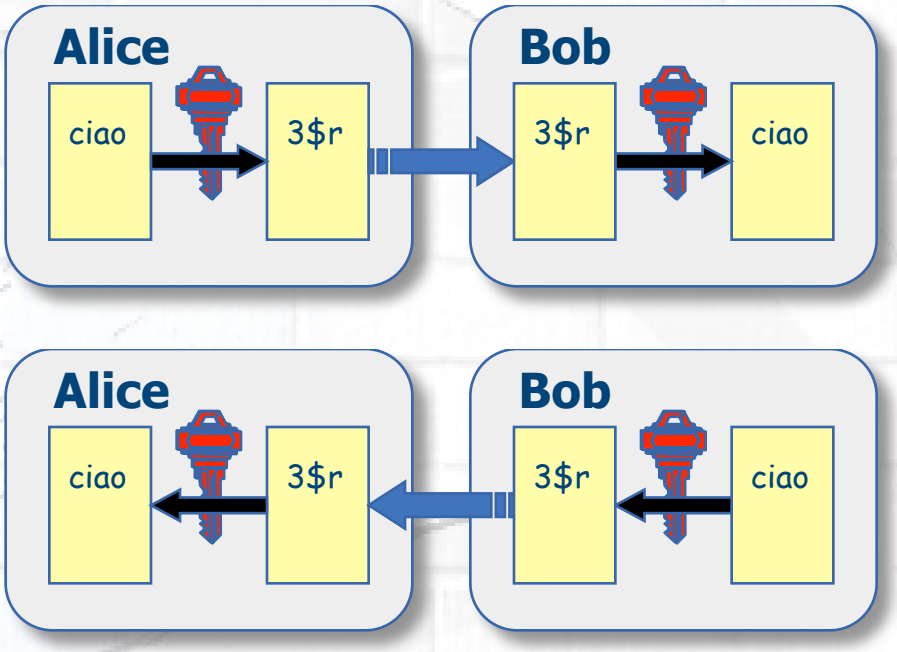
- **Algorithms**

  Symmetric: $K_1 = K_2$

  Asymmetric: $K_1 \neq K_2$

Symmetric Algorithms

Same key is for encryption and decryption



Advantages:

Fast, easy to understand

Disadvantages:

how distribute the keys?

Number of keys is $O(n^2)$

Examples:

DES

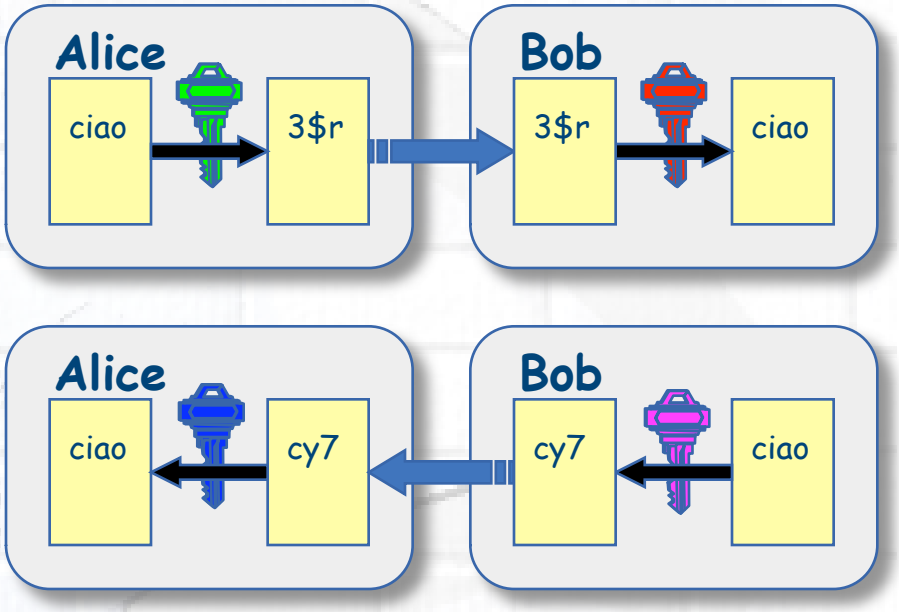
Blowfish

Kerberos

Public key algorithms

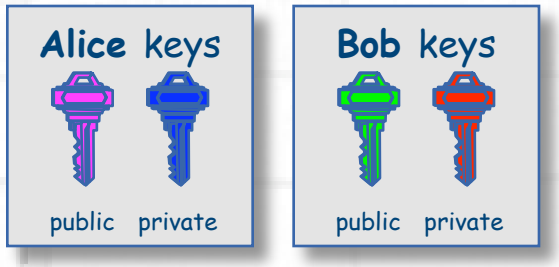
Every user has two keys: a *private* and a *public one* :

- it is *impossible* to derive the private key from the public one;
- a message encrypted by one key can be decrypted **only** by the other one



No exchange of secrets is necessary

- the sender cyphers using the *public* key of the receiver;
- the receiver decrypts using his *private* key;
- the number of keys is O(n).



Examples:

RSA (1978)

Digital Signature

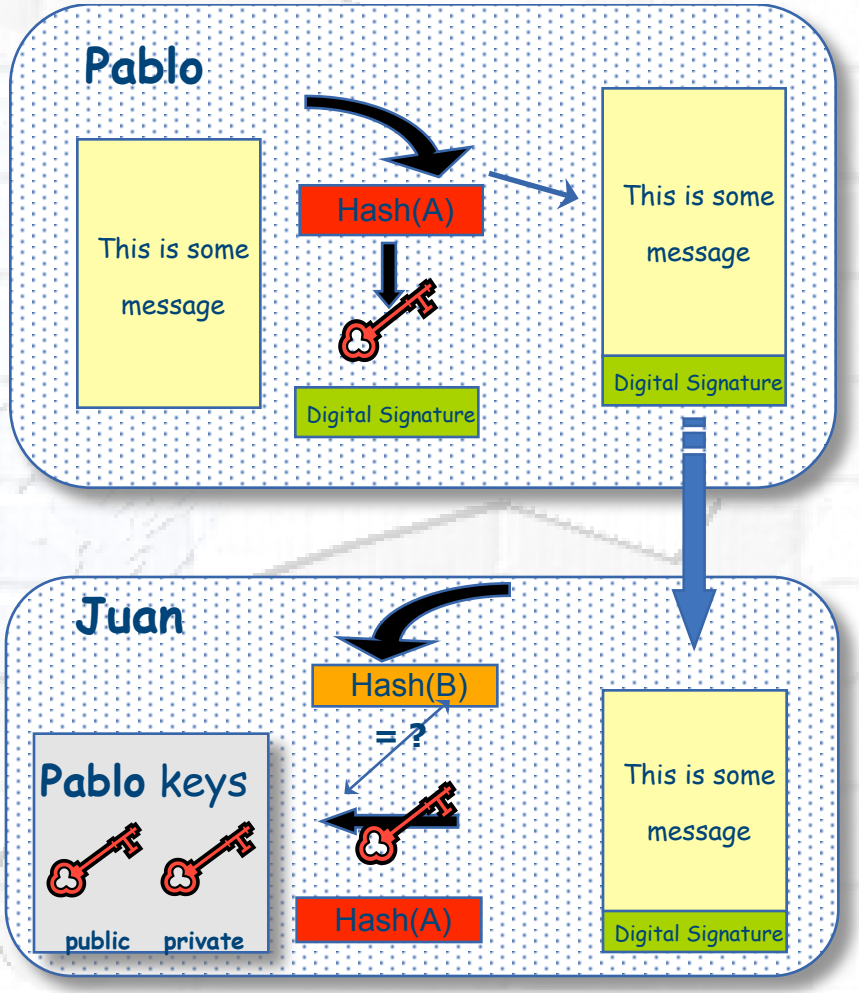
Pablo **calculates the** *hash* **of the message (with a one-way hash function)**

Pablo **encrypts the hash using his** *private* **key: the encrypted hash is the** *digital signature*.

Pablo **sends the signed message to** John.

Juan **calculates the hash of the message and** *verifies* **it with A, decyphered with Pablo's** *public* **key.**

If hashes equal: message wasn't modified; Pablo **cannot repudiate it.**



Digital Certificates

- **Pablo's digital signature is safe if:**
  1. Pablo's private key is not compromised
  2. Juan knows Pablo's public key
- **How can Juan be sure that Pablo's public key is really Pablo's public key and not someone else's?**
  - A *third party* guarantees the correspondence between public key and owner's identity.
  - Both A and B must trust this third party
- **Two models:**
  - X.509: hierarchical organization;
  - PGP: "web of trust".

Certification authority

The “third party” is called *Certification Authority* (CA)

- Issue Digital Certificates (containing public key and owner’s identity) for users, programs and machines (signed by the CA)
- Check identity and the personal data of the requestor
  - Registration Authorities (RAs) do the actual validation
- CA’s periodically publish a list of compromised certificates
  - **Certificate Revocation Lists** (**CRL**): contain all the revoked certificates yet to expire

Content of an X509 certificate

Structure of a X.509 certificate

An X.509 Certificate contains

owner's public key;

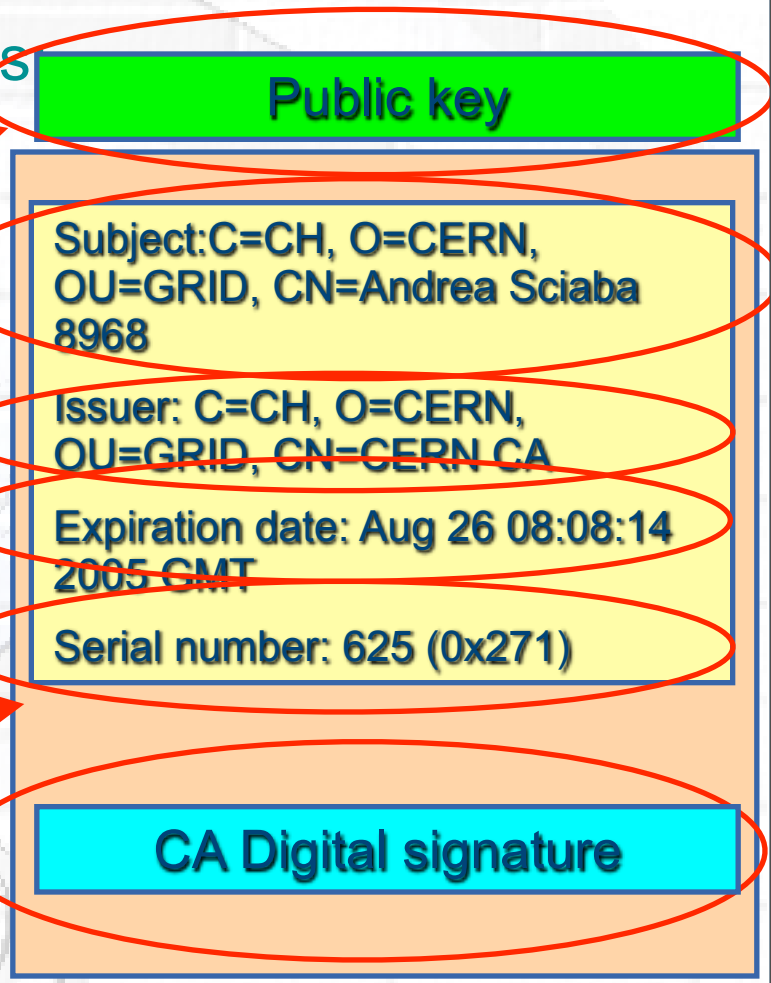
identity of the owner;

info on the CA;

time of validity;

Serial number;

digital signature of the CA

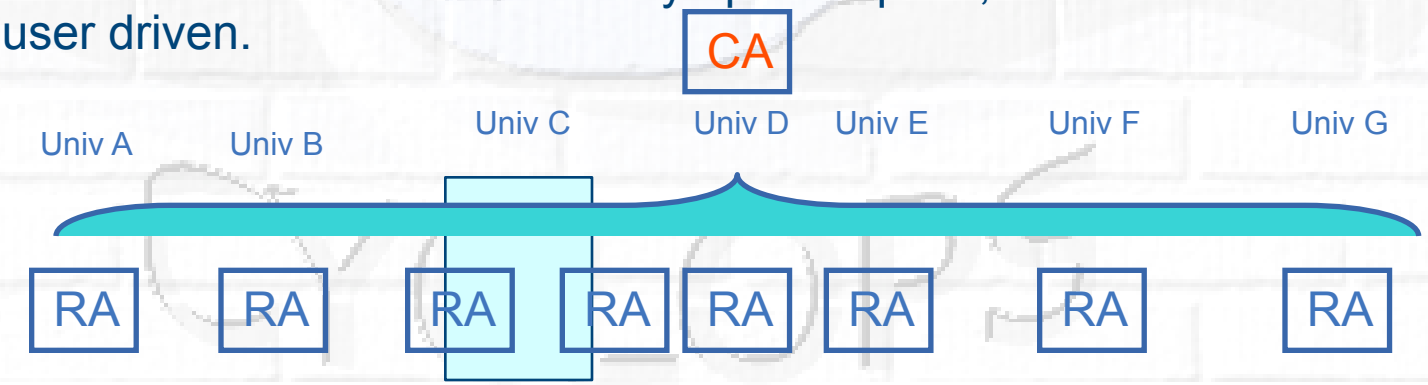


More on CA's

- In the grid world one single CA usually covers a predefined geographic region or administrative domain:
  - Organization
  - Country
  - A set of countries
- A common trust domain for grid computing has been created to join the several existing certification authorities into a single authentication domain and thus enabling sharing of grid resources worldwide.
  - The International Grid Trust Federation (IGTF) has been created to coordinate and manage this trust domain.
  - IGTF is divided in three Policy Management Authorities (PMAs) covering the Asia Pacific, Europe and Americas.

Classic Profile of a CA : RA

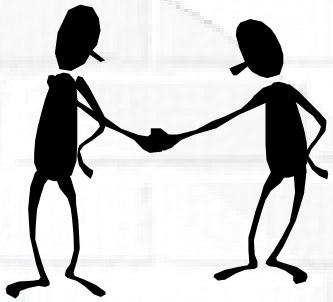
- A network of subordinated RAs (**Registration Authority**) is necessary to perform the identity verification of the subjects
- The RAs will be created at the level of the organizations or at the level of departments:
  - Operating at university or research centre wide level (more difficult)
  - Operating at the level of a department or group
  - The CA can also operate an RA but don't forget that the physical presence of the subject is required for identity verification
  - It is fine to have more than one RA per university or research centre if they are operating for different departments
- The RAs should be created only upon request, their creation should be user driven.



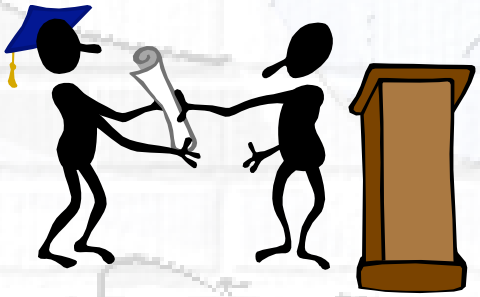
Classic profile of a CA

How to obtain a certificate

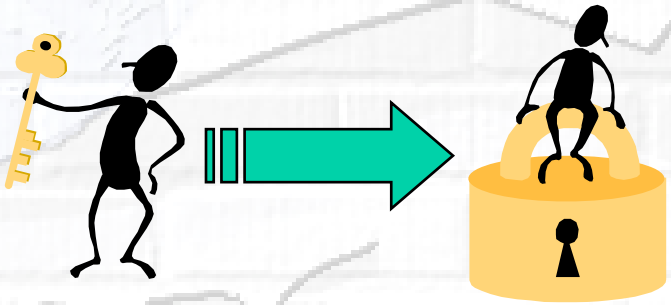
A certificate request is performed



The user identify is confirmed by the RA



The certificate is issued by the CA



The certificate is used as a key to access the grid

Request of an INFN certificate

- Before requesting a personal certificate, user must be authenticated by a Registration Authority. In detail
  - User goes phisically to RA which verifies his identity (https://security.fi.infn.it/CA/RA/ shows all the INFN RA)
  - RA opens URL: https://security.fi.infn.it/cgi-bin/RAvfy.pl and fills it with user's data: name, surname, e-mail; finally, a random number is generated and communicated to user.
  - If needed, user with its browser downloads INFN CA public cert
  - within 48 hours from the communication of the code by the RA, the user submit the certificate request using the same values used before by the RA
    - https://security.fi.infn.it/CA/mgt/restricted/ucert.php
  - if everything is ok, with 48 working hours, user will receive instruction on how to download its personal certificate; **he/she must use the same browser used for the request**

Certificate management

- Import your certificate in your browser
  - If you received a .pem certificate you need to convert it to PKCS12
  - Use *openssl* command line (available in each UI)
    - `openssl pkcs12 -export -in usercert.pem -inkey userkey.pem -out my_cert.p12 -name 'My Name'`
- Most of other CA's:
  - You receive already a PKCS12 certificate (can import it directly into the web browser)
  - For future use, you will need *usercert.pem* and *userkey.pem* in a directory ~/.globus on your UI
  - Export the PKCS12 cert to a local dir on UI and use again *openssl:*
    - `openssl pkcs12 -nocerts -in my_cert.p12 -out userkey.pem`
    - `openssl pkcs12 -clcerts -nokeys -in my_cert.p12 -out usercert.pem`

X509 proxy

- GSI extension to X.509 Identity Certificates
  - signed by the normal end entity cert (or by another proxy).
- Enables single sign-on
- Support some important features
  - Delegation
  - Mutual authentication
- Has a limited lifetime (minimized risk of “compromised credentials”)
- It is created by the `grid-proxy-init` command:

```
% grid-proxy-init
Enter PEM pass phrase: ******
```

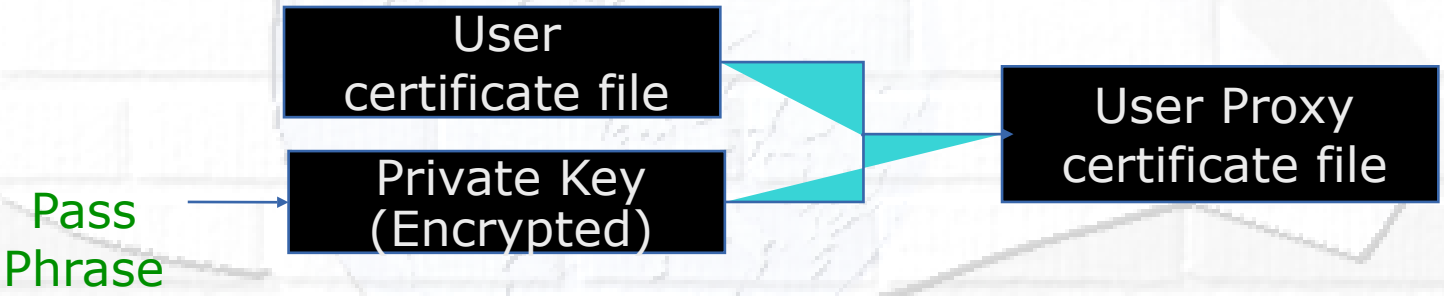
  - Options for `grid-proxy-init`:
    - -hours <lifetime of credential>
    - -bits <length of key>

How proxies are created ?

User enters pass phrase, which is used to decrypt private key.

Private key is used to sign a proxy certificate with its own, new public/private key pair.

User's private key not exposed after proxy has been signed



Proxy placed in /tmp

the private key of the Proxy is *not* encrypted:

stored in local file: must be readable **only** by the owner;

proxy lifetime is short (typically 12 h) to minimize security risks.

NOTE: *No* network traffic!

Proxy again …

- grid-proxy-init ≡ “login to the Grid”
- To “logout” you have to destroy your proxy:
  - **`grid-proxy-destroy`**
  - This does *NOT* destroy any proxies that were delegated from this proxy.
  - You cannot revoke a remote proxy
  - Usually create proxies with short lifetimes
- To gather information about your proxy:
  - **`grid-proxy-info`**
  - Options for printing proxy information

    -subject -issuer
    -type -timeleft
    -strength

Delegation

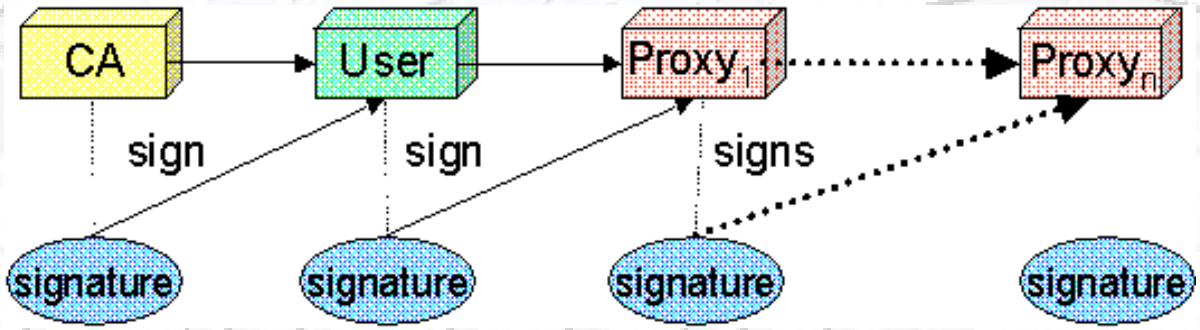
Delegation = remote creation of a (second level) proxy credential

New key pair generated remotely on server

Client signs proxy cert and returns it

Allows remote process to authenticate on behalf of the user

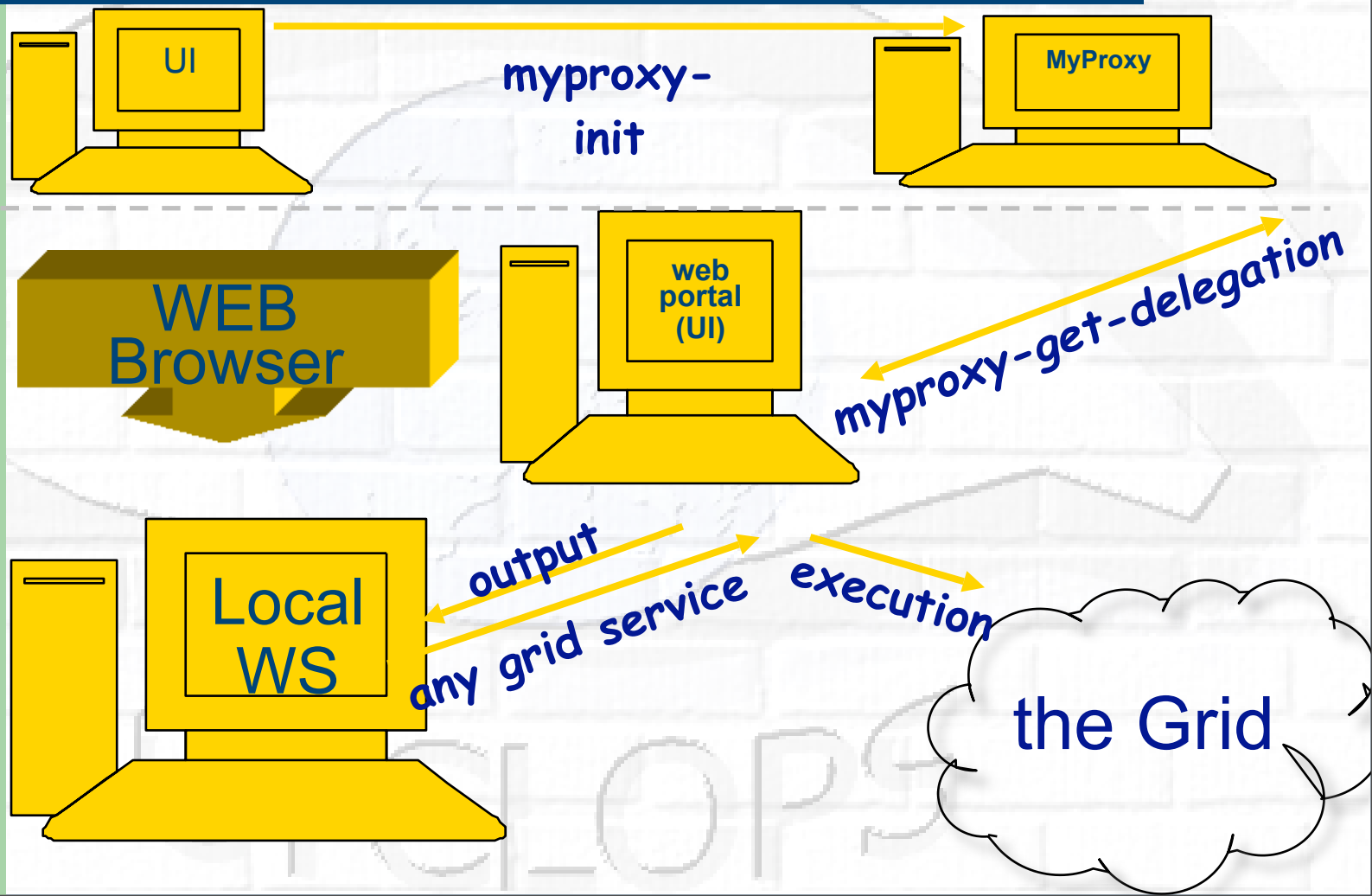
Remote process “impersonates” the user



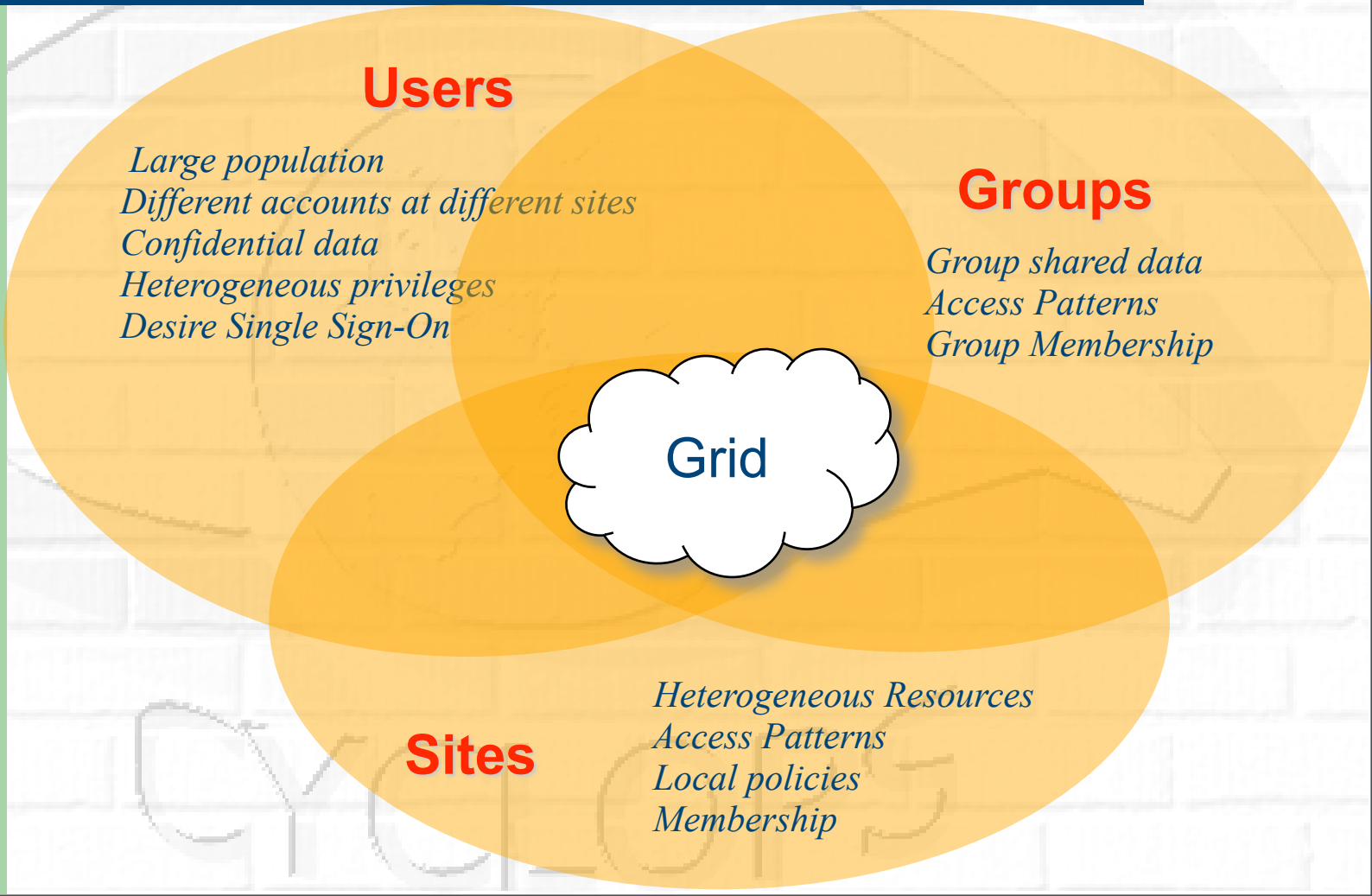
MyProxy

- Proxy has limited lifetime (default is 12 h)
  - Bad idea to have longer proxy
- However, a grid task might need to use a proxy for q longer time
  - Grid jobs in HEP Data Challenges on LCG last up to 2 days
- myproxy server:
  - Allows to create and store a long term proxy certificate:
  - `myproxy-init -s` <host_name>
    - -s: <host_name> specifies the hostname of the myproxy server
  - `myproxy-info`
    - Get information about stored long living proxy
  - `myproxy-get-delegation`
    - Get a new proxy from the MyProxy server
  - `myproxy-destroy`
  - Check out the myproxy-xxx - - help option
- A dedicated service on the RB can renew automatically the proxy
  - contacting myproxy server

Grid authentication with MyProxy



GRID Security: the players



Pre-VOMS authorization

- Grid users MUST belong to virtual organizations
  - What we previously called “groups”
  - Sets of users belonging to a collaboration
  - User must sign the usage guidelines for the VO
  - You will be registered in the VO server (wait for notification)
- VOs maintained a list of their members on a LDAP Server
  - The list was downloaded by grid machines to map user certificate subjects to local “pool” accounts

```
...
"/C=CH/O=CERN/OU=GRID/CN=Simone Campana 7461" .dteam
"/C=CH/O=CERN/OU=GRID/CN=Andrea Sciaba 8968" .cms
"/C=CH/O=CERN/OU=GRID/CN=Patricia Mendez Lorenzo-ALICE" .alice
...
```

  - Sites decide which vos to accept
    - /etc/grid-security/grid-mapfile

Evolution of VO management

Before VOMS

- User is authorised as a member of a single VO
- All VO members have same rights
- Gridmapfiles are updated by VO management software: map the user's DN to a local account
- grid-proxy-init – derives proxy from certificate – the "single sign-on to the grid"

VOMS

User can be in multiple VOs
Aggregate rights

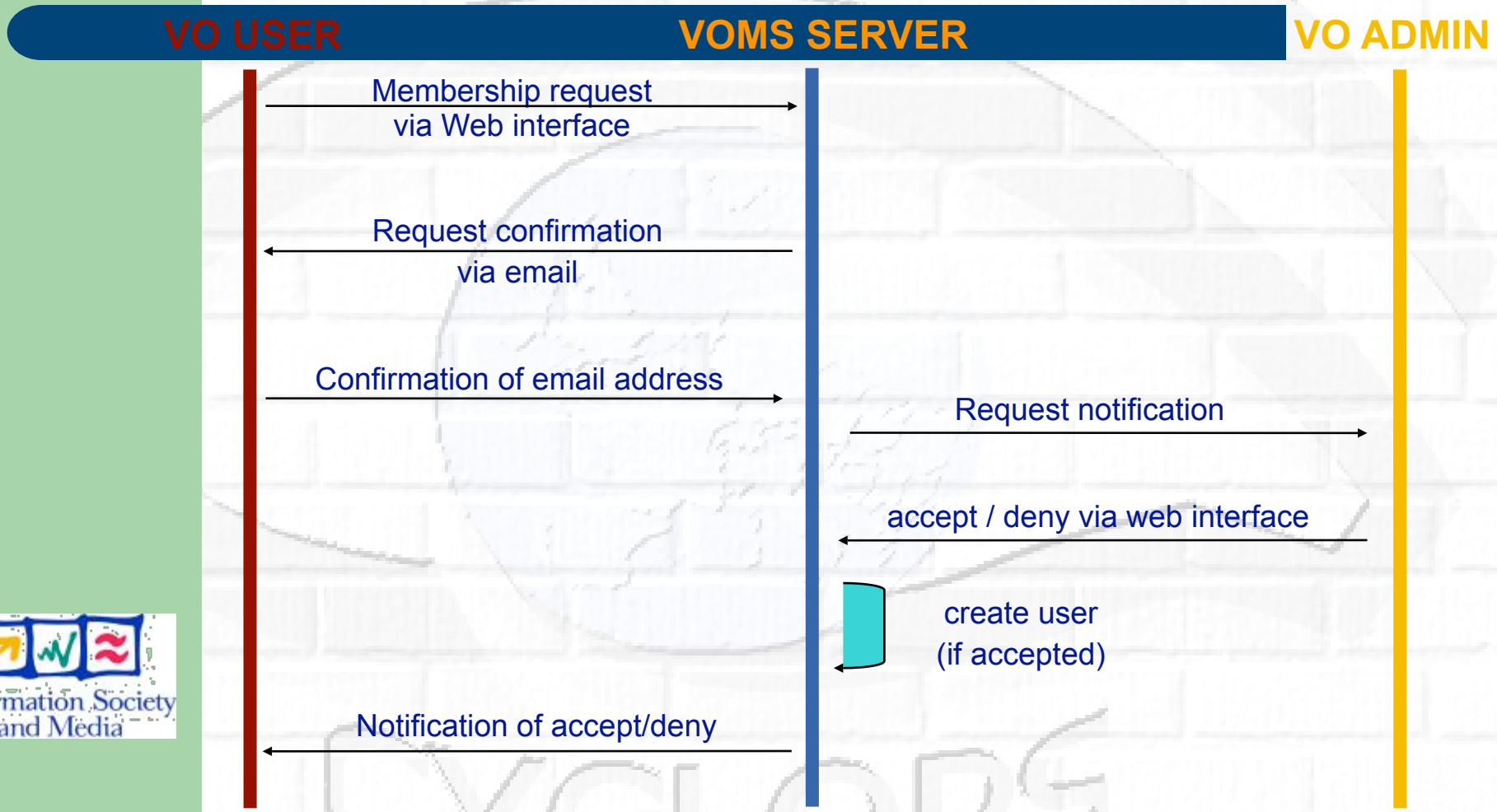
VO can have groups
Different rights for each
Different groups of experimentalists
…
Nested groups

VO has roles
Assigned to specific purposes
E,g. system admin
When assume this role

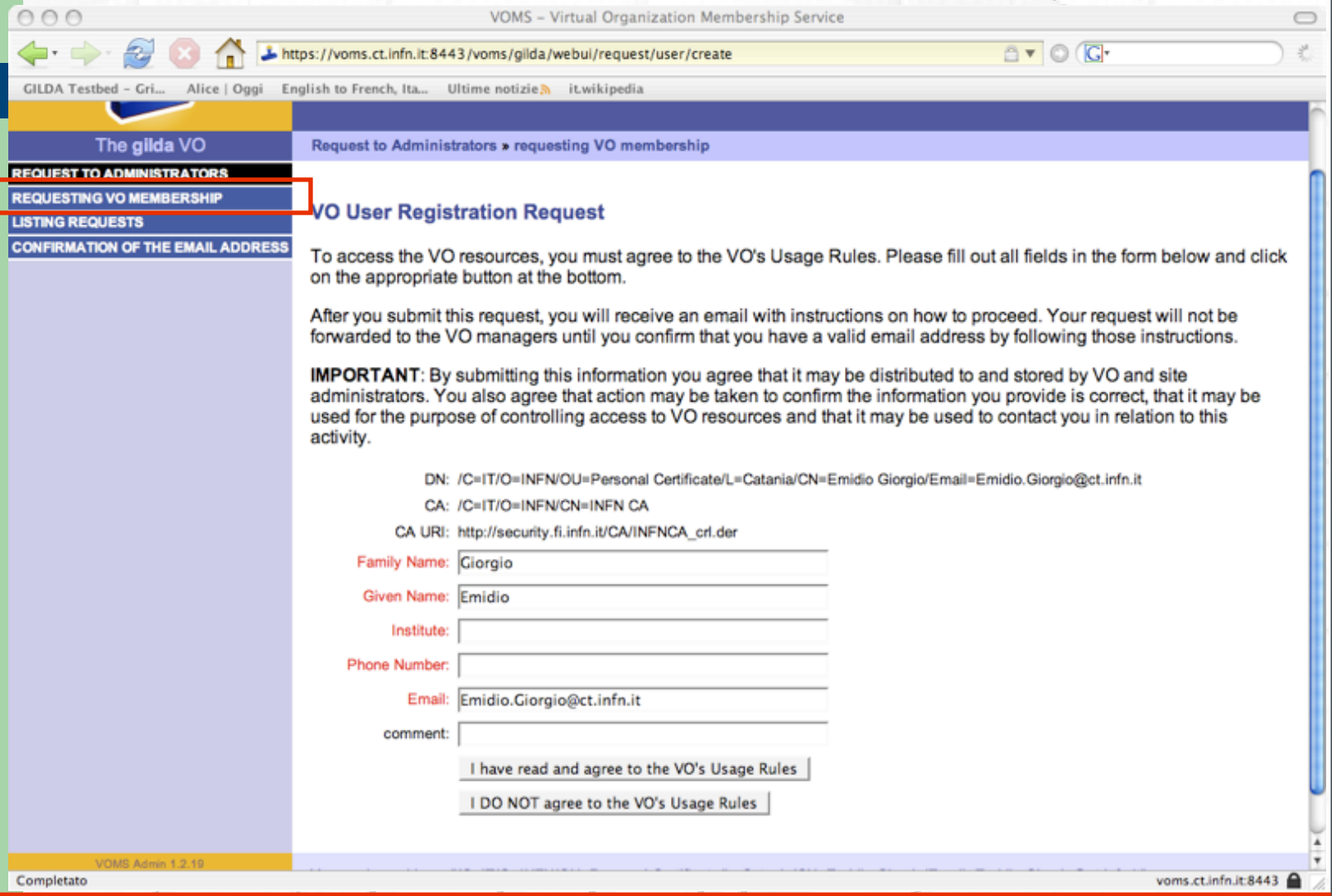
Proxy certificate carries the additional attributes

voms-proxy-init

Registration process



GILDA VOMS
(https://voms.ct.infn.it:8443/voms/gilda/)



New registrations at: https://voms.ct.infn.it:8443/voms/gilda/webui/request/user/create

The VOMS client

- Virtual Organization Membership Service
- Extends the proxy with info on VO membership, group, roles
- Fully compatible with Globus Toolkit
- Each VO has a database containing group membership, roles and capabilities informations for each user
- User contacts voms server requesting his authorization info
- Server send authorization info to the client, which includes them in a proxy certificate

```
[glite-tutor] /home/giorgio > voms-proxy-init --voms gilda
Your identity: /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/
CN=Emidio Giorgio/Email=emidio.giorgio@ct.infn.it
Enter GRID pass phrase:
Your proxy is valid until Mon Jan 30 23:35:51 2006
Creating temporary proxy.....................................Done
Contacting  voms.ct.infn.it:15001 [/C=IT/O=INFN/OU=Host/L=INFN
Catania/CN=voms.ct.infn.it] "gilda"
Creating proxy ..................................... Done
Your proxy is valid until Mon Jan 30 23:35:51 2006
```

FQAN and AC

FQAN : short for Fully Qualified Attribute Name, is what VOMS uses to express membership and other authorization info

Groups membership, roles and capabilities may be expressed in a format that bounds them together
<group>/Role=[<role>][/Capability=<capability>]

```
[glite-tutor] /home/giorgio > voms-proxy-info -fqan
/gilda/Role=NULL/Capability=NULL
/gilda/tutors/Role=NULL/Capability=NULL
```

FQAN are included in an **Attribute Certificate**

Attribute Certificates are used to bind a set of attributes (like membership, roles, authorization info etc) with an identity

AC are digitally signed

VOMS uses AC to include the attributes of a user in a proxy

VOMS and AC

Server creates and sign an AC containing the FQAN requested by the user, if applicable

AC is included by the client in a well-defined, non critical, extension assuring compatibility with GT-based mechanism

```
/home/giorgio > voms-proxy-info -all
subject   : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio Giorgio/
Email=emidio.giorgio@ct.infn.it/CN=proxy
issuer    : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio Giorgio/
Email=emidio.giorgio@ct.infn.it
identity  : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio Giorgio/
Email=emidio.giorgio@ct.infn.it
type      : proxy
strength  : 512 bits
path      : /tmp/x509up_u513
timeleft  : 11:59:52
=== VO gilda extension information ===
VO        : gilda
subject   : /C=IT/O=GILDA/OU=Personal Certificate/L=INFN/CN=Emidio Giorgio/
Email=emidio.giorgio@ct.infn.it
issuer    : /C=IT/O=INFN/OU=Host/L=INFN Catania/CN=voms.ct.infn.it
attribute : /gilda/tutors/Role=NULL/Capability=NULL
attribute : /gilda/Role=NULL/Capability=NULL
timeleft  : 11:59:45
```

Groups

- The number of users of a VO can be very high:
  - E.g. the experiment ATLAS has 2000 member
- Make VO manageable by organizing users in groups:

  Examples:
  - VO GILDA
    - Group Catania
      - INFN
        - Group Barbera
      - University
    - Group Padua
  - VO GILDA
    - /GILDA/TUTORS — can write to normal storage
    - /GILDA/STUDENT — only write to volatile space
- Groups can have a hierarchical structure, indefinitely deep

Roles

- Roles are specific roles a user has and that distinguishes him from others in his group:
  - Software manager
  - VO-Administrator
- Difference between roles and groups:
  - Roles have no hierarchical structure – there is no sub-role
  - Roles are not used in 'normal operation'
    - Not added to the proxy by default when running *voms-proxy-init*
    - Can be added to the proxy if needed when running *voms-proxy-init*
- Example:
  - User Emidio has the following membership
    - VO=gilda, Group=tutors, Role=SoftwareManager
  - During normal operation the role is not taken into account, e.g. Emidio can work as a normal user
  - For special things he can obtain the role "Software Manager"
  - Explicit request with the appropriate option to command

LCAS and LCMAPS

At resources level, authorization info are extracted from the proxy and processed by *LCAS* and *LCMAPS*

Local Centre Authorization Service (LCAS)

Checks if the user is authorized

Checks if the user is banned at the site

Checks if at that time the site accepts jobs

Local Credential Mapping Service (LCMAPS)

Maps grid credentials to local credentials (eg. UNIX uid/gid, AFS tokens, etc.)

Map also VOMS group and roles (full support of FQAN)

```
"/VO=cms/GROUP=/cms"                          .cms
"/VO=cms/GROUP=/cms/prod"                     .cmsprod
"/VO=cms/GROUP=/cms/prod/ROLE=manager"        .cmsprodman
```

CYCLOPS

- In this course all security stuffs have already been setup for you
  - you have an account on GILDA UI
  - You have a generic certificate released by GILDA CA
  - You have been inserted on GILDA VO
  - You have been inserted in /generic-users
  - You have been assigned to Role GenericRole and /generic-users/Role=GenericRole (see the hands on for the differences)
- Just exercise !
- https://grid.ct.infn.it/twiki/bin/view/GILDA/AuthenticationAuthorization
- https://grid.ct.infn.it/twiki/bin/view/GILDA/VomsClientGroupRole
- https://grid.ct.infn.it/twiki/bin/view/GILDA/MyProxyUse

Information Society and Media

Access to the UI

- ssh bolognaXX@glite-tutor.ct.infn.it
- password GridBOLXX
- XX=[01,15]
- Attention to capital letters !
- certificate passphrase :BOLOGNA (the same for all users)

References

- Grid
  - LCG Security:
    - http://proj-lcg-security.web.cern.ch/proj-lcg-security/
  - Globus Security Infrastructure:
    - http://www.globus.org/security/
  - VOMS: http://infnforge.cnaf.infn.it/projects/voms
  - CA: http://www.tagpma.org/
- Background
  - GGF Security: http://www.gridforum.org/security/
  - IETF PKIX charter:
    - http://www.ietf.org/html.charters/pkix-charter.html
  - PKCS: http://www.rsasecurity.com/rsalabs/pkcs/index.html