



Authorization and Authentication in gLite

Diego Scardaci – INFN Catania

1st BioMed Grid School - Varenna, 14-18/05/2007

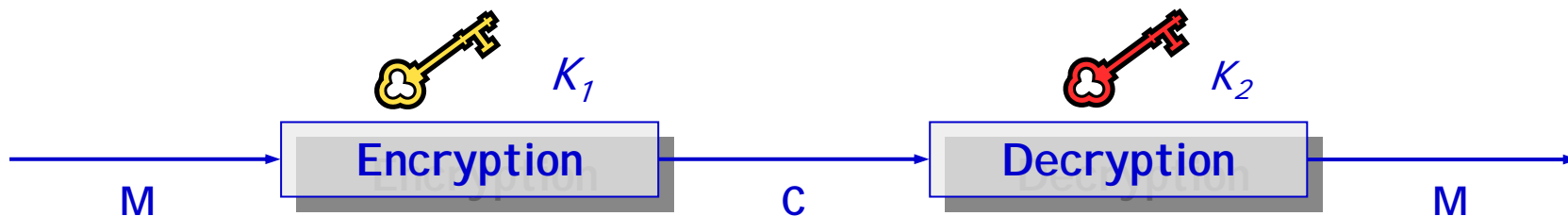




- ▶ **Glossary**
- ▶ **Encryption**
 - Symmetric algorithms
 - Asymmetric algorithms: PKI
- ▶ **Certification Authorities**
 - Digital Signatures
 - X509 certificates
- ▶ **Grid Security**
 - Basic concepts
 - Grid Security Infrastructure
 - Proxy certificates
 - Commands used in UI
- ▶ **Virtual Organization**
 - Concept of VO and authorization
 - VOMS, LCAS, LCMAPS
- ▶ **References**



- ▶ **Principal**
 - An entity: a user, a program, or a machine
- ▶ **Credentials**
 - Some data providing a proof of identity
- ▶ **Authentication**
 - Verification of the identity for an end-entity
- ▶ **Authorization**
 - Map an entity 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 during the transmission
- ▶ **Non-repudiation**
 - Impossibility of denying the authenticity of a digital signature

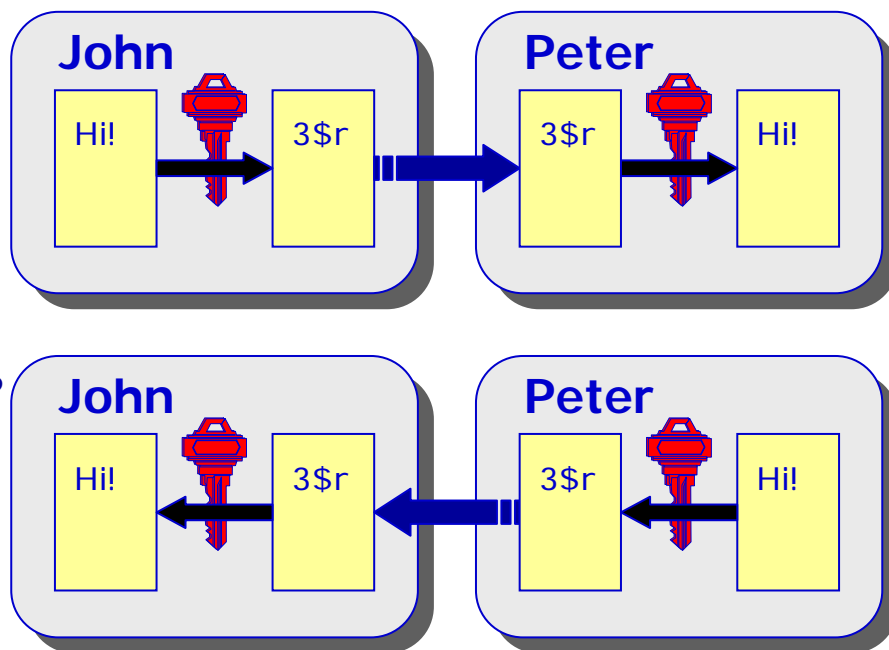


- ▶ To implement the security infrastructure, cryptography uses mathematical algorithms that provide important building blocks.
- ▶ Corresponding definitions for the above symbols:
 - **Plaintext: M**
 - **Cyphertext: C**
 - **Encryption with key K_1 : $E_{K_1}(M) = C$**
 - **Decryption with key K_2 : $D_{K_2}(C) = M$**
- ▶ Algorithms
 - **Symmetric: $K_1 = K_2$**
 - **Asymmetric: $K_1 \neq K_2$**



Symmetric Algorithms

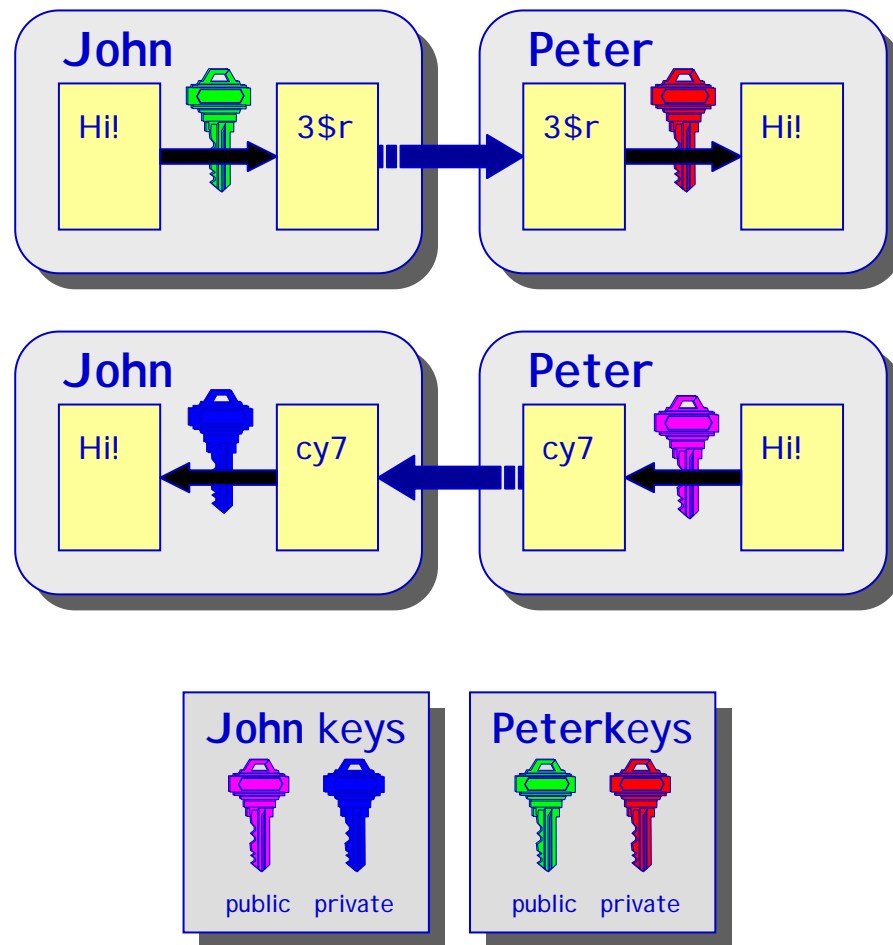
- ▶ The same key is used for encryption and decryption (no public key, only secret keys available.)
- ▶ Advantages:
 - Fast
- ▶ Disadvantages:
 - Exchange of secret keys needed:
 - how to distribute the keys?
 - the number of keys is $O(n^2)$
- ▶ Examples:
 - DES
 - 3DES
 - AES





Public Key Algorithms (Asymmetric)

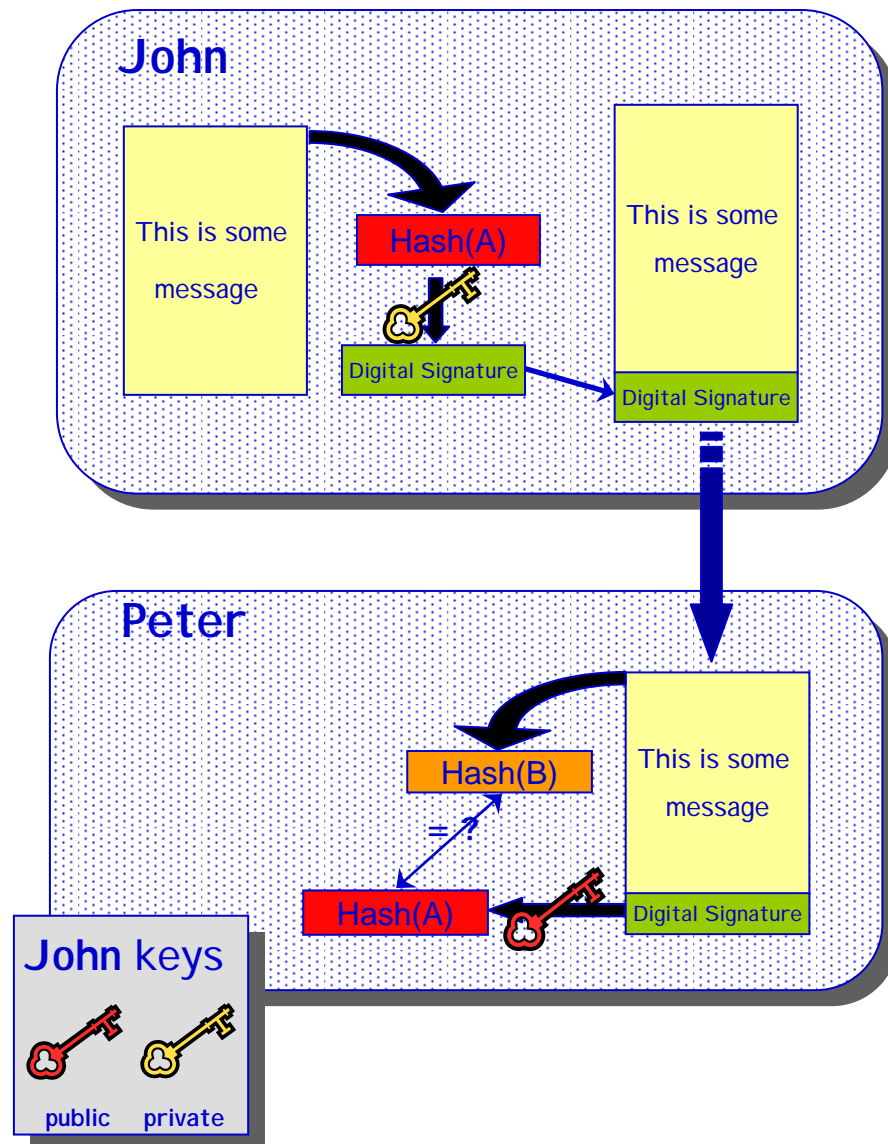
- ▶ Every user has two keys: one *private* (*secret*) and one *public*:
 - it is *impossible* to derive the private key from the public one;
 - a message encrypted by one key can be decrypted **only** by the other one.
- ▶ No exchange of private key is needed.
 - the sender cyphers using the *public* key of the receiver;
 - the receiver decrypts using his own *private* key;
 - the number of keys is $O(n)$.
- ▶ Examples:
 - **RSA** (1978)





Digital Signature

- ▶ **John** calculates the **hash** of the message (with a one-way hash function)
- ▶ **John** encrypts the hash using his **private** key: the encrypted hash is the **digital signature**.
- ▶ **John** sends the signed message to **Peter**.
- ▶ **Peter** calculates the hash of the message and **verifies** it with A, decyphered with **Peter's public** key.
- ▶ If two hashes equal: message wasn't modified; **John** cannot repudiate it.





- ▶ **John's digital signature is safe if:**
 1. John's private key is not compromised
 2. Peter knows John's public key
- ▶ **How can Peter be sure that John's public key is really John's public key and not someone else's?**
 - A *third party* guarantees the correspondence between public key and owner's identity.
 - Both John and Peter must trust this third party
- ▶ **Two models proposed to build trust:**
 - X.509: hierarchical organization (**used in Grid**)
 - PGP: "web of trust". (person to person)



X.509 and Certification Authorities

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

Responsibilities of CA:

- ▶ Issue **Digital Certificates** (containing public key and owner's identity) for users, programs and machines
- ▶ Check identity and the personal data of the requestor
 - Registration Authorities (RAs) do the actual validation
- ▶ Revoke certificates in case of a compromise
- ▶ Renew certificates in case of expiration
- ▶ Periodically publish a list of revoked certificates through web repository
 - **Certificate Revocation Lists** (CRL): contain all the revoked certificates

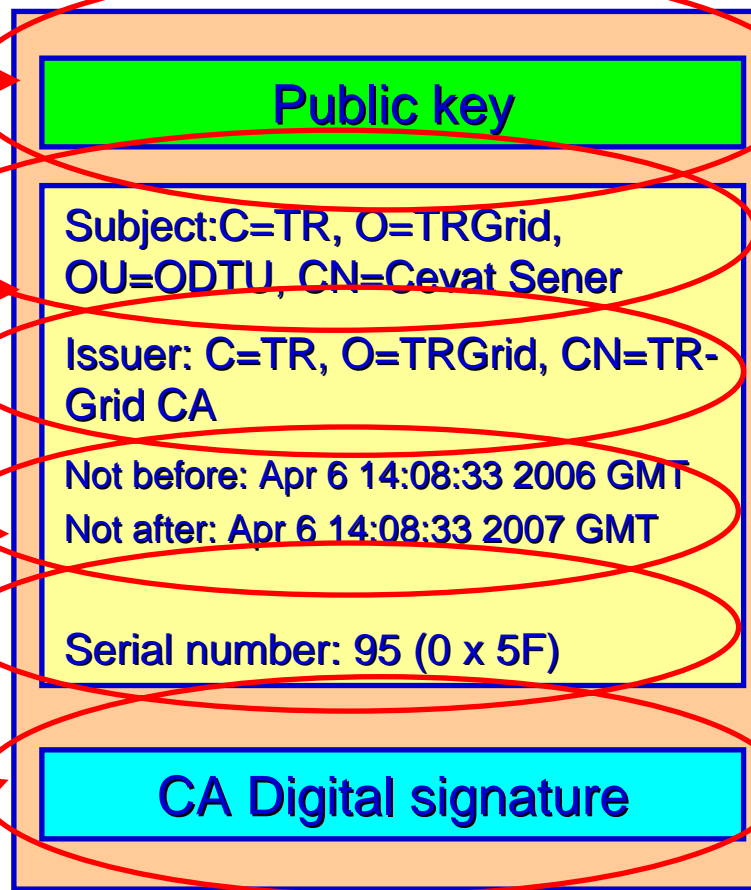
- ▶ **CA certificates are self-signed**



► An X.509 Certificate contains:

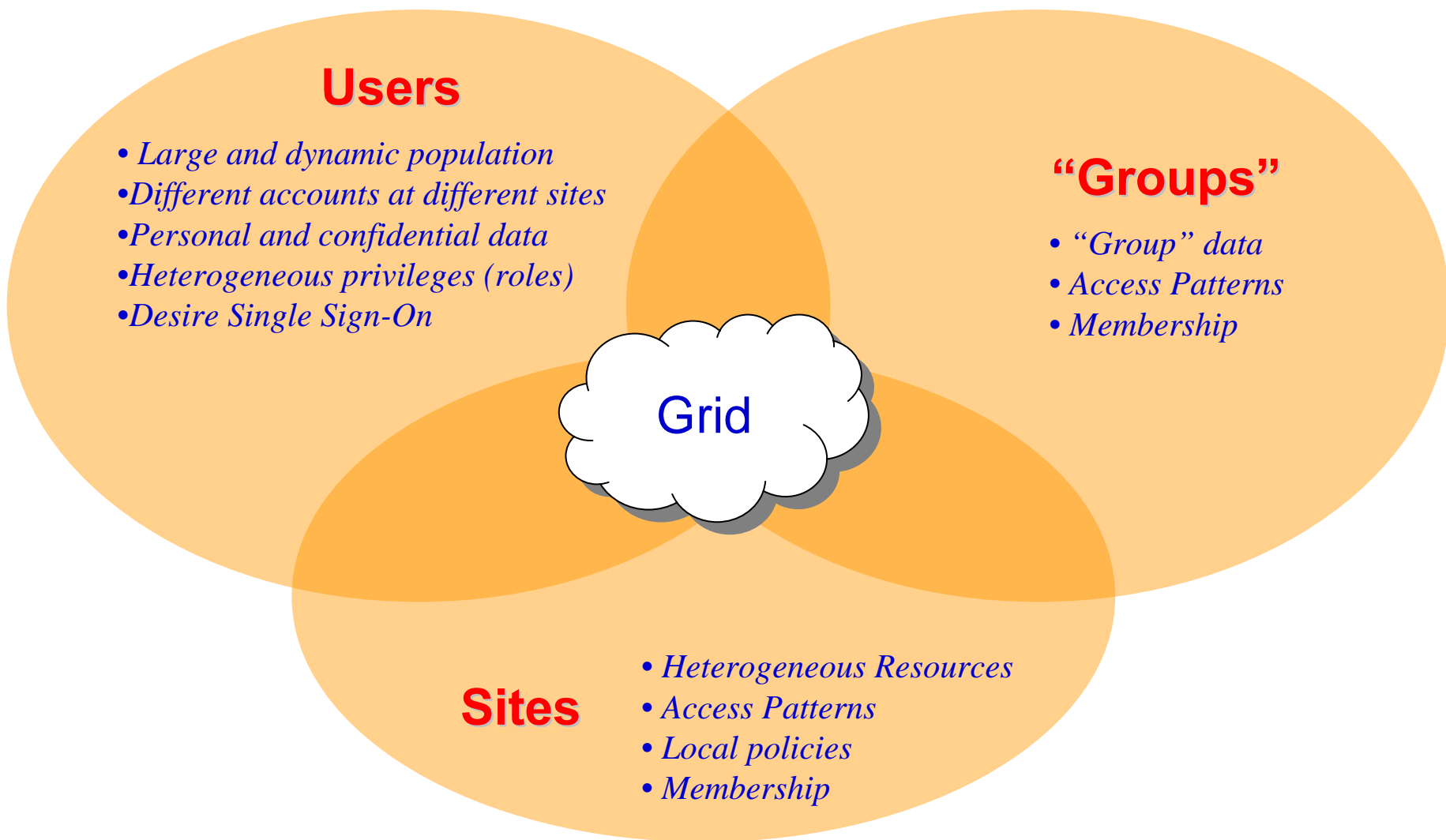
- owner's public key;
- identity of the owner (DN);
- info on the CA;
- time of validity;
- Serial number;
- digital signature of the CA

Structure of a X.509 certificate





GRID Security: Components





The Grid Security Infrastructure (GSI)

BioinfoGRID

Based on X.509 PKI:

- ▶ every user/host/service has an X.509 certificate;
- ▶ certificates are signed by trusted (by the local s
- ▶ every Grid authentic
 1. John s
 2. Peter certifi
 3. Peter string
 4. John e with h
 5. John s Peter
 6. Peter decryp
 7. Peter compares the decrypted string with the original challenge
 8. If they match, Peter verifies John's identity and John can not repudiate it.

John

Peter

John's certificate

VERY IMPORTANT

Private keys must be stored only by

owners:

in **protected** places

AND

in **encrypted** form

private key

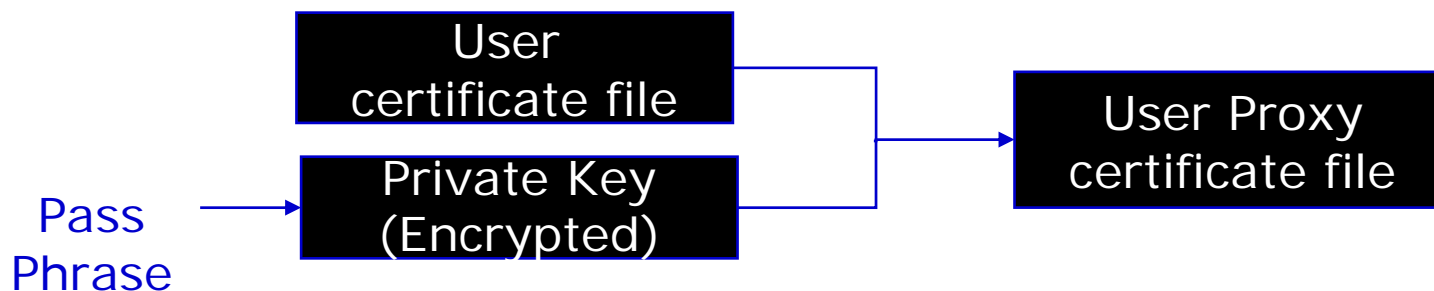
key



- ▶ **Proxy: GSI extension to X.509 Identity Certificates**
 - signed by the normal end entity cert (or by another proxy).
- ▶ **It enables single sign-on.**
- ▶ **It supports some important features:**
 - Delegation
 - Mutual authentication
- ▶ **It 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>`
 - `-help`



- ▶ 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!**

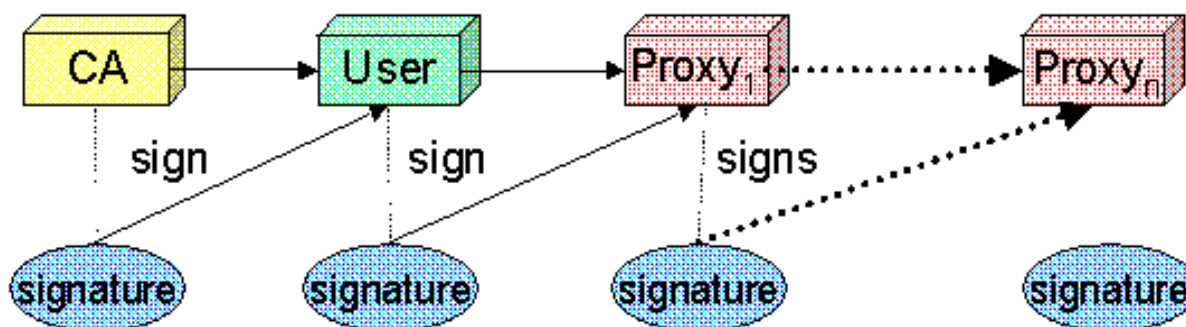


- ▶ **grid-proxy-init** \equiv “login to the Grid”
- ▶ **To “logout” you have to destroy your proxy:**
 - `grid-proxy-destroy`
 - This does *NOT* destroy any proxies that were delegated from this proxy.
 - You cannot revoke a remote proxy

- ▶ **To gather information about your proxy:**
 - `grid-proxy-info`
 - Options for `grid-proxy-info`:
 - subject -issuer
 - type -timeleft
 - strength -help



- ▶ **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





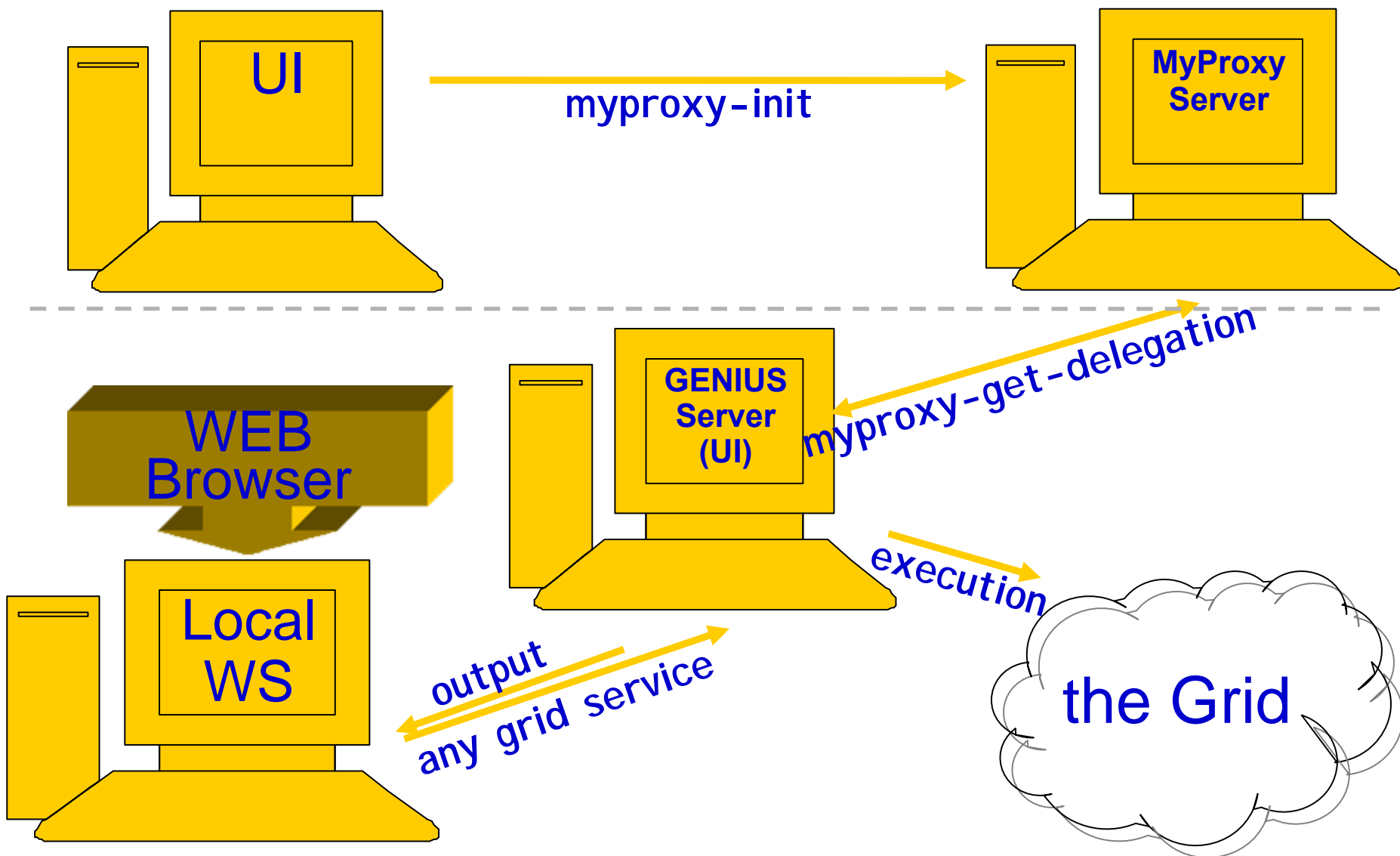
Long term proxy --> Myproxy

BioinfoGRID

- ▶ **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 much longer time**
 - Grid jobs in HEP 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 for more information
- ▶ **A dedicated service on the RB can renew automatically the proxy**
- ▶ **File transfer services in gLite validates user request and eventually renew proxies**
 - contacting myproxy server



Grid authentication with MyProxy





- ▶ **Grid users MUST belong to virtual organizations**
 - It was called “groups” previously.
 - It defines sets of users belonging to a collaboration
 - User must sign the usage guidelines for the VO
 - You will be registered in the VO-LDAP server (wait for notification)
 - List of supported VOs:
 - https://lcg-registrar.cern.ch/virtual_organization.html
- ▶ **VOs maintain a list of their members on a LDAP Server**
 - The list is downloaded by grid machines to map user certificate subjects to local “pool” accounts
 - Sites decide which VOs to support
 - /etc/grid-security/grid-mapfile

```
• "/C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Birsen Omay" .seegrid  
• "/C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Hakan Bayindir" .trgridb  
• "/C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Onur Temizsoylu" .dteam
```



▶ 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 information for each user
- User contacts voms server requesting his authorization info
- Server sends authorization info to the client, which includes them in a proxy certificate

```
asli@levrek:~$ voms-proxy-init --voms trgrida
Your identity: /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Asli Zengin
Enter GRID pass phrase:
Creating temporary proxy ..... Done
Contacting voms.ulakbim.gov.tr:15051 [/C=TR/O=TRGrid/OU=TUBITAK-
ULAKBIM/CN=voms.ulakbim.gov.tr] "trgrida" Done
Creating proxy ..... Done
Your proxy is valid until Sat Jul 1 04:02:30 2006
```



- ▶ 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>]`

```
asli@levrek:~$ voms-proxy-info -fqan  
/trgrida/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
- ▶ ACs are digitally signed
- ▶ VOMS uses AC to include the attributes of a user in a proxy certificate



- ▶ Server creates and signs 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**

```
asli@levrek:~$ voms-proxy-info -all
```

```
subject      : /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Asli Zengin/CN=proxy
issuer       : /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Asli Zengin
identity     : /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Asli Zengin
type         : proxy
strength     : 512 bits
path         : /tmp/x509up_u1134
timeleft     : 11:58:08
VO           : trgrida
subject      : /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=Asli Zengin
issuer       : /C=TR/O=TRGrid/OU=TUBITAK-ULAKBIM/CN=voms.ulakbim.gov.tr
attribute    : /trgrida/Role=NULL/Capability=NULL
timeleft     : 11:58:08
```



- ▶ **At resources level, authorization info is extracted from the proxy and processed by LCAS and LCMAPS**
- ▶ **Local Centre Authorization Service (LCAS)**
 - Checks if the user is authorized (currently using the grid-mapfile)
 - 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=dteam/GROUP=/dteam" dteam`
 - `"/VO=eumed/GROUP=/eumed/ROLE=SoftwareManager" eumed`
 - `"/VO=eumed/GROUP=/eumed" eumed`



▶ User certificate files:

- Certificate: **X509_USER_CERT**
(default: `$HOME/.globus/usercert.pem`)
- Private key: **X509_USER_KEY**
(default: `$HOME/.globus/userkey.pem`)
- Proxy: **X509_USER_PROXY**
(default: `/tmp/x509up_u<id>`)

▶ Host certificate files:

- Certificate: **X509_HOST_CERT**
(default: `/etc/grid-security/hostcert.pem`)
- Private key: **X509_HOST_KEY**
(default: `/etc/grid-security/hostkey.pem`)



- ▶ **Trusted certification authority certificates:**
 - **X509_CERT_DIR**
(default: `/etc/grid-security/certificates`)

- ▶ **Voms server public keys**
 - **X509_VOMS_DIR**
(default: `/etc/grid-security/vomsdir`)



▶ Grid

- LCG Security: <http://proj-lcg-security.web.cern.ch/proj-lcg-security/>
- LCG Registration: <http://lcg-registrar.cern.ch/>
- Globus Security: <http://www.globus.org/security/>
- VOMS: <http://infnforge.cnaf.infn.it/projects/voms>
- IGTF for trusted CAs: <http://www.gridpma.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>