



Enabling Grids for E-scienceE

Security, Authorisation and Authentication

Gergely Sipos

MTA SZTAKI

sipos@sztaki.hu

With thanks for some slides to EGEE and Globus colleagues

www.eu-egee.org

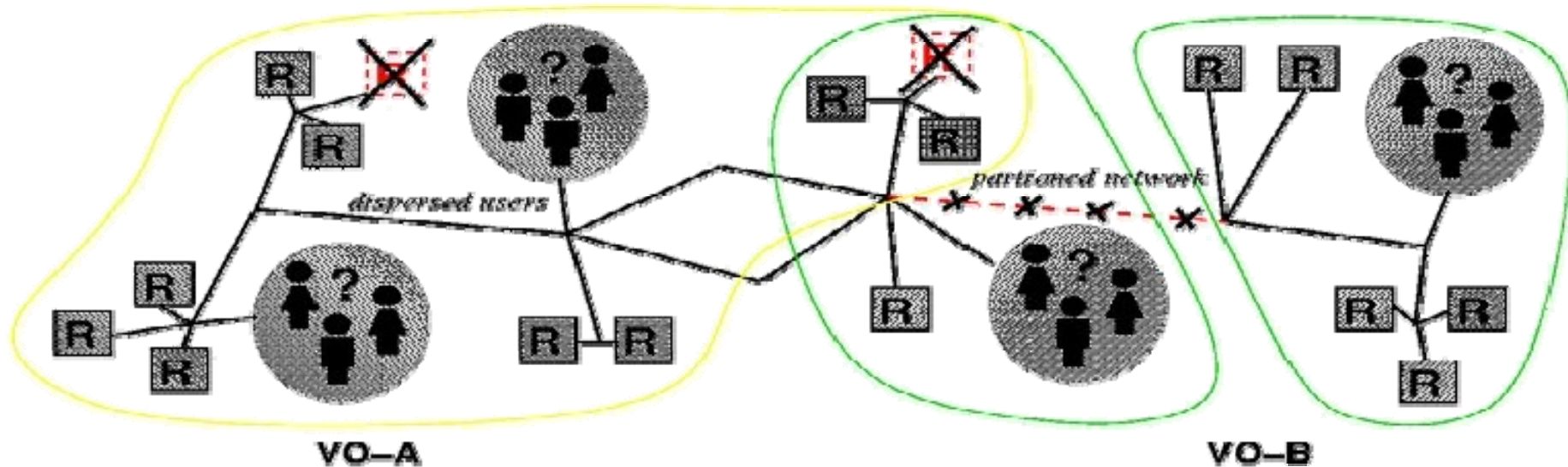


***The Grid problem is to enable
“coordinated resource sharing and
problem solving in dynamic, multi-
institutional virtual organizations.”***

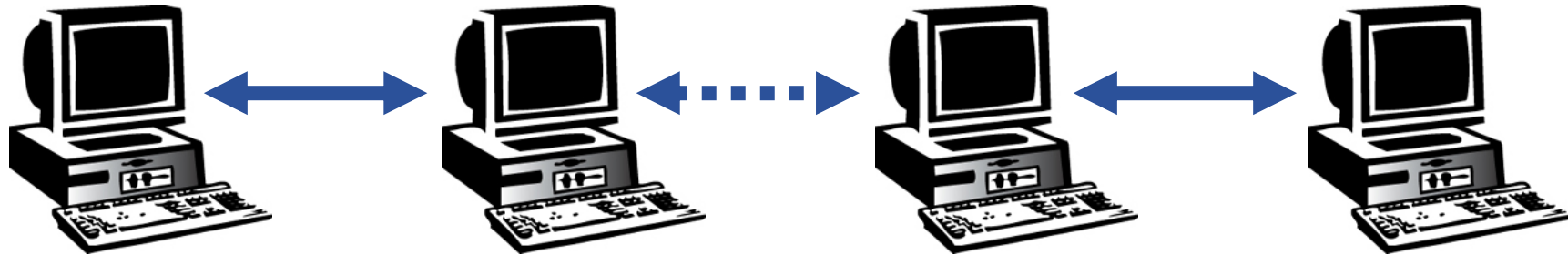
From "The Anatomy of the Grid" by Ian Foster et al.

- So Grid Security is security to enable VOs
- What is needed in terms of security for a VO?

Virtual Organization (VO) Concept



- VO for each application or workload
- Carve out and configure resources for a particular use and set of users
- The more dynamic the better...



User

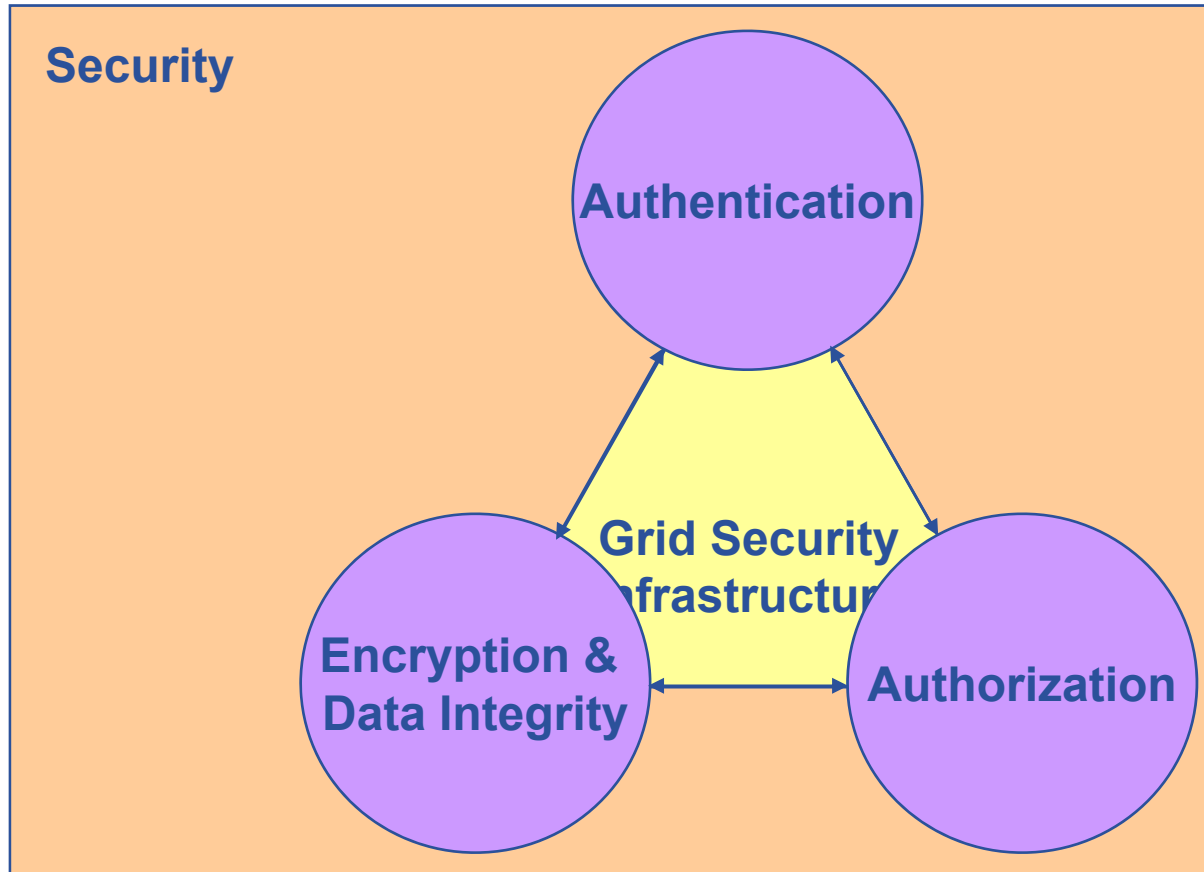
Resource

- How can the members of the VO identified?
- Who does belong to a VO? Who does not?
- How does the machine in the VO know who its current user is?
- How are rights controlled?
- How does a user securely access the Resource without having an account with username and password on the machines in between or even on the Resource?

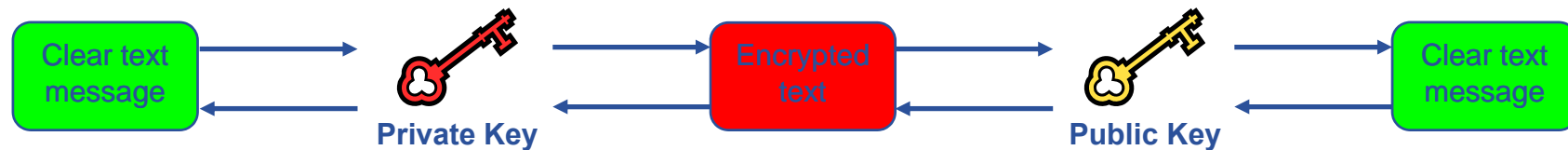
Authentication: how is identity of user/site communicated?

Authorisation: what can a user do?

- **Launch attacks to other sites**
 - Large distributed farms of machines, perfect for launching a Distributed Denial of Service attack.
- **Illegal or inappropriate data distribution and access sensitive information**
 - Massive distributed storage capacity ideal for example, for swapping movies.
 - Growing number of users have data that must be private – biomedical imaging for example
- **Damage caused by viruses, worms etc.**
 - Highly connected infrastructure means worms could spread faster than on the internet in general.



- **Asymmetric encryption...**



- **.... and Digital signatures ...**

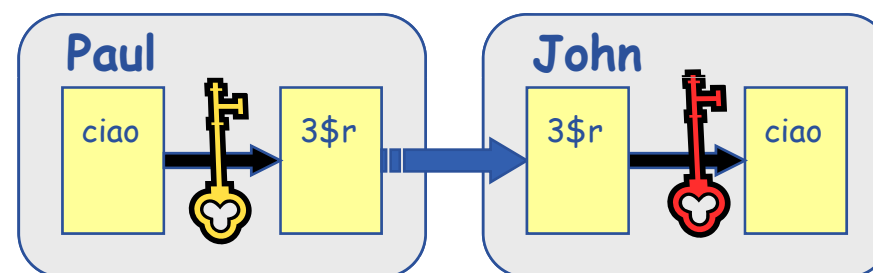
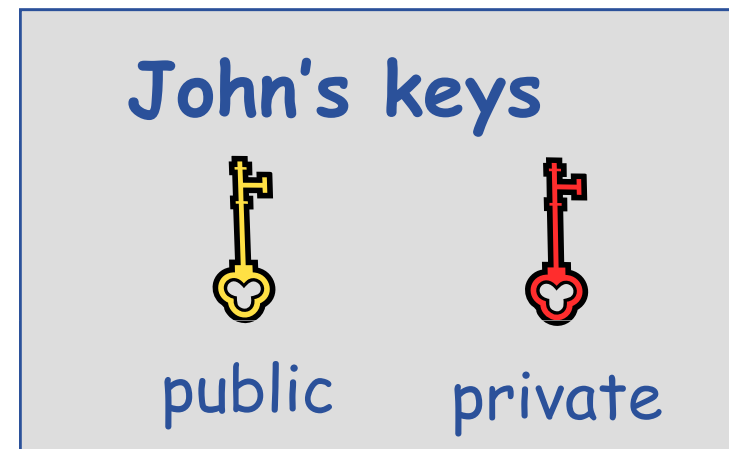
- A hash derived from the message and encrypted with the signer's private key
- Signature is checked by decrypting with the signer's public key

- **Are used to build trust**

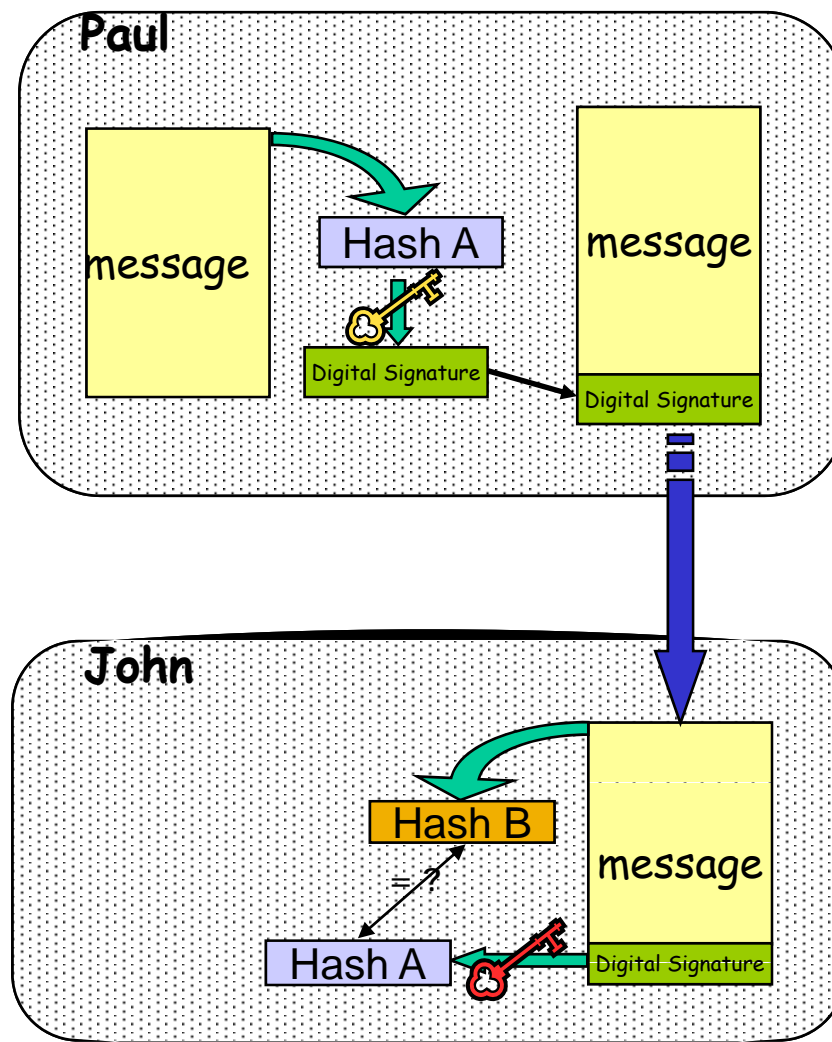
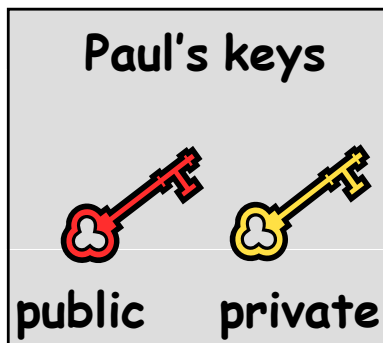
- That a user / site is who they say they are
- And can be trusted to act in accord with agreed policies

- Every entity that wants to join a VO (user/machine/software) has two keys: one *private* 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.

- **Concept - simplified version:**
 - Public keys are exchanged
 - The sender encrypts using receiver's public key
 - The receiver decrypts using their private key;



- Paul calculates the *hash* of the message: a 128 bit value based on the content of the message
- Paul encrypts the hash using his *private* key: the encrypted hash is the *digital signature*.
- Paul sends the signed message to John.
- John calculates the hash of the message → Hash B
- Decrypts A with Paul's *public* key → Hash A
- If hashes equal:
 1. hash B is from Paul's private key;
 2. message wasn't modified;

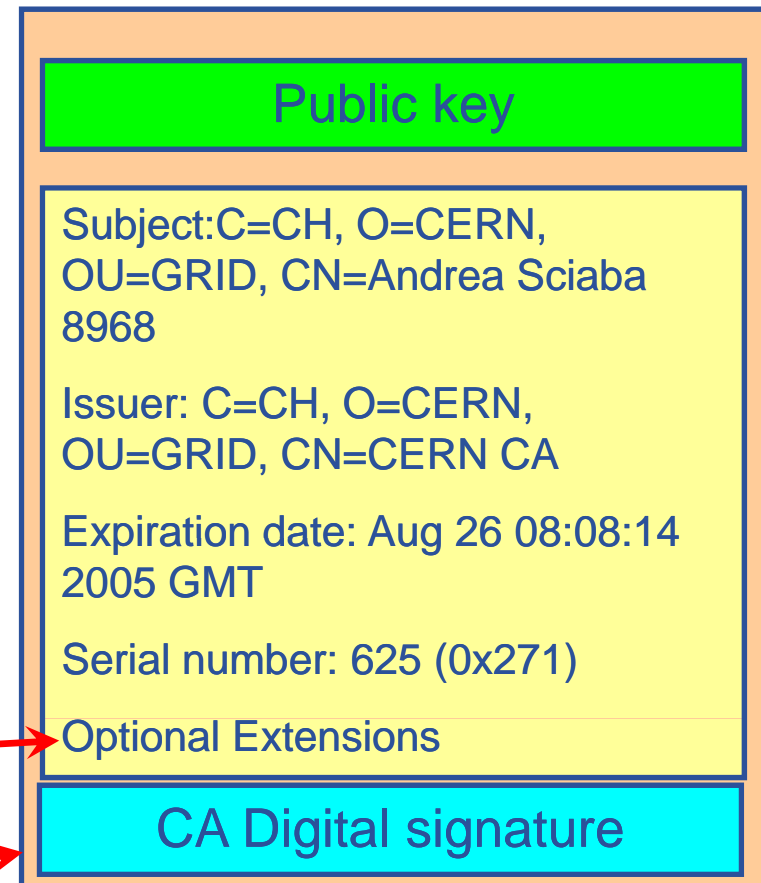


- How can John be sure that Paul's public key is really Paul's public key and not someone else's?
 - A *third party* signs a certificate that binds the public key and Paul's identity.
 - Both John and Paul trust this third party

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

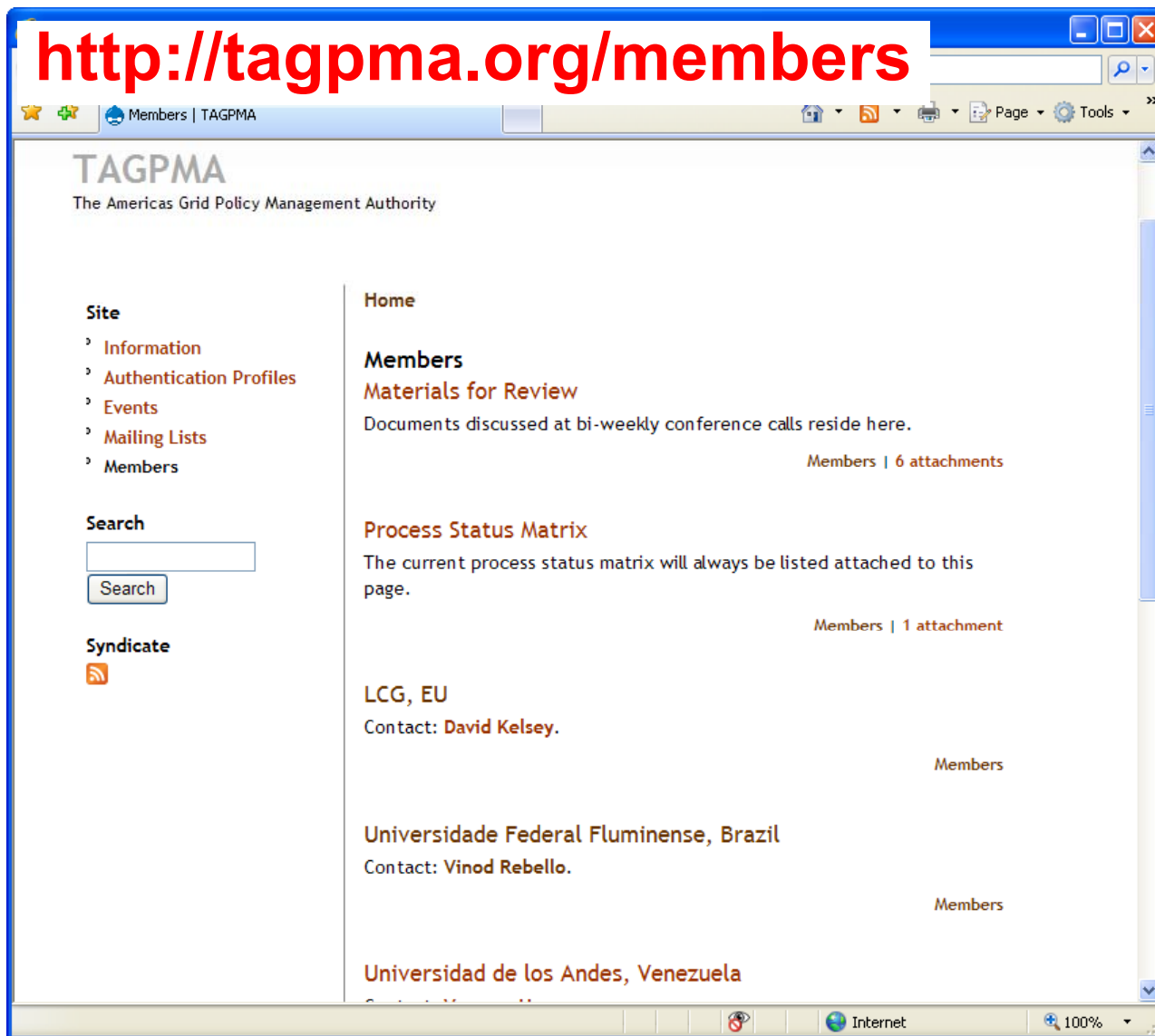
- **An X.509 Certificate contains:**

- owner's public key; →
- identity of the owner; →
- info on the CA; →
- time of validity; →
- Serial number; →
- Optional extensions →



- digital signature of the CA →

- **User's identity has to be certified by one of the national *Certification Authorities (CAs)***
- **Resources are also certified by CAs**
- **CAs are mutually recognized**
<http://www.gridpma.org/>
- **CAs can establish a number of people “registration authorities” RAs**
 - Personal visit to the nearest RA instead of the CA



Universidade Federal Fluminense, Brazil
 Contact: Vinod Rebello

Universidad de los Andes, Venezuela
 Contact: Vanessa Hamar

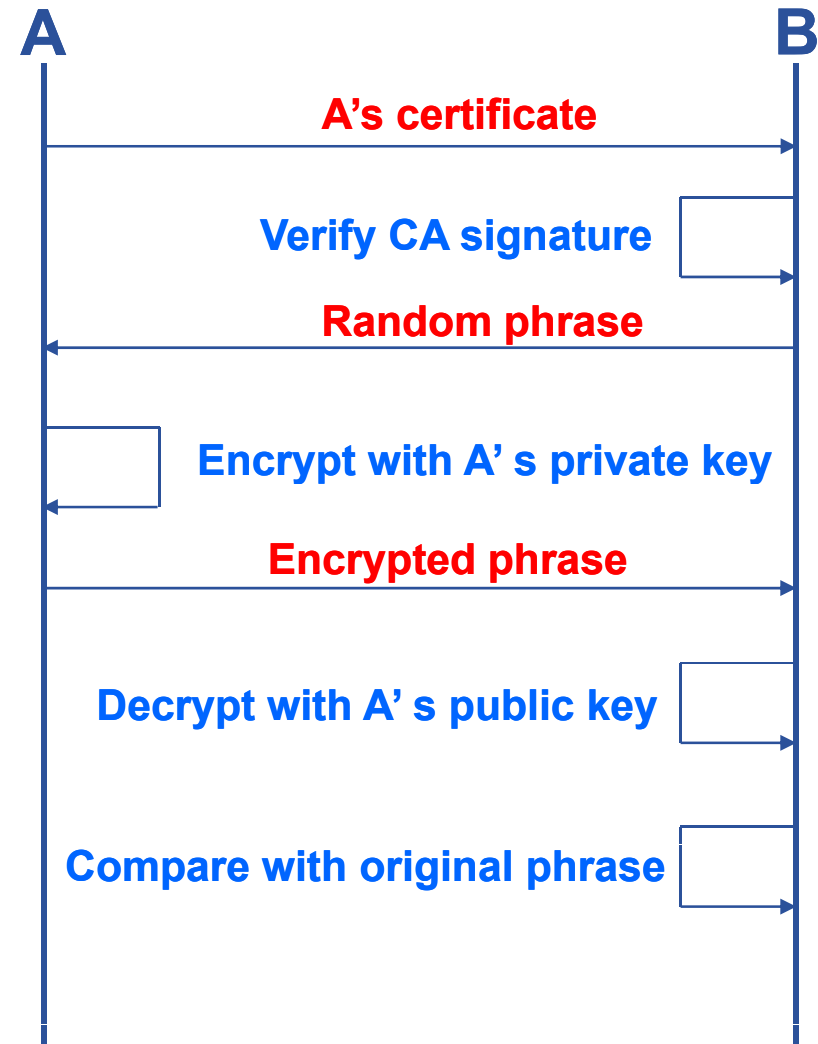
Red Universitaria Nacional, Chile
 Contact: Claudia Inostroza

...

EELA, EU & Latin America
 Contact: Jorge Gomes

Based on X.509 PKI:

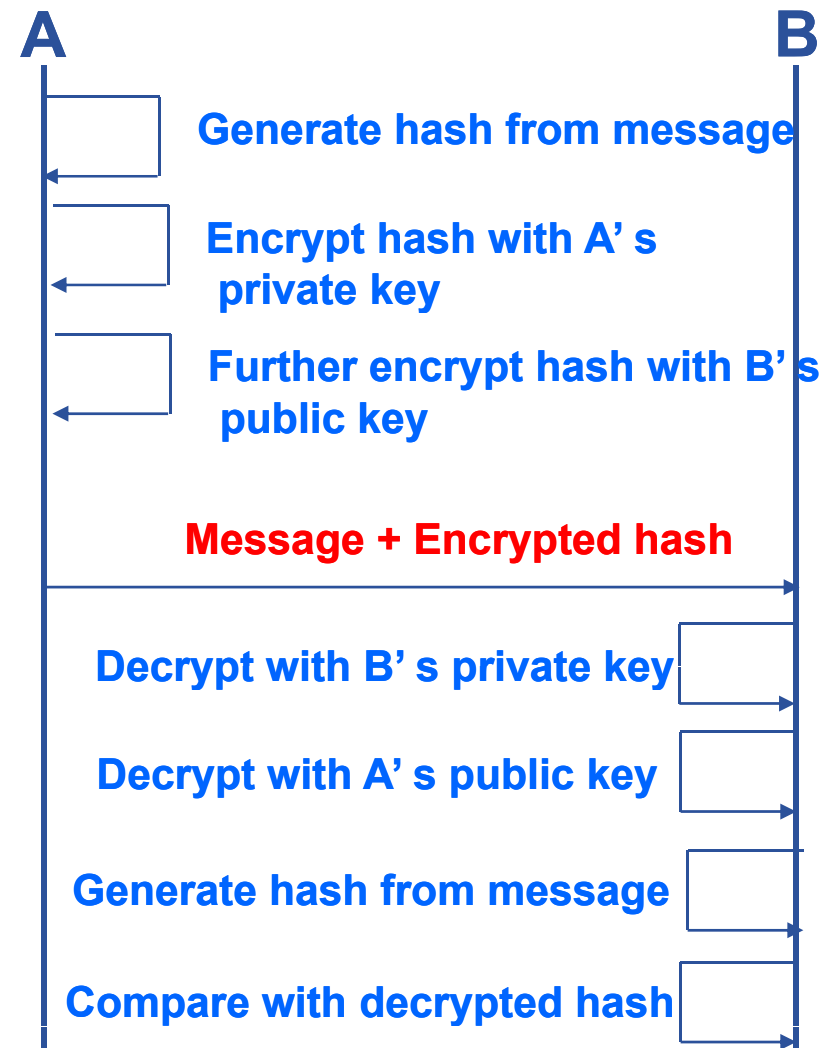
- every Grid transaction is mutually authenticated:
 1. A sends his certificate;
 2. B verifies signature in A's certificate using CA public certificate;
 3. B sends to A a challenge string;
 4. A encrypts the challenge string with his private key;
 5. A sends encrypted challenge to B
 6. B uses A's public key to decrypt the challenge.
 7. B compares the decrypted string with the original challenge
 8. If they match, B verified A's identity and A can not repudiate it.
 9. Repeat for A to verify B's identity



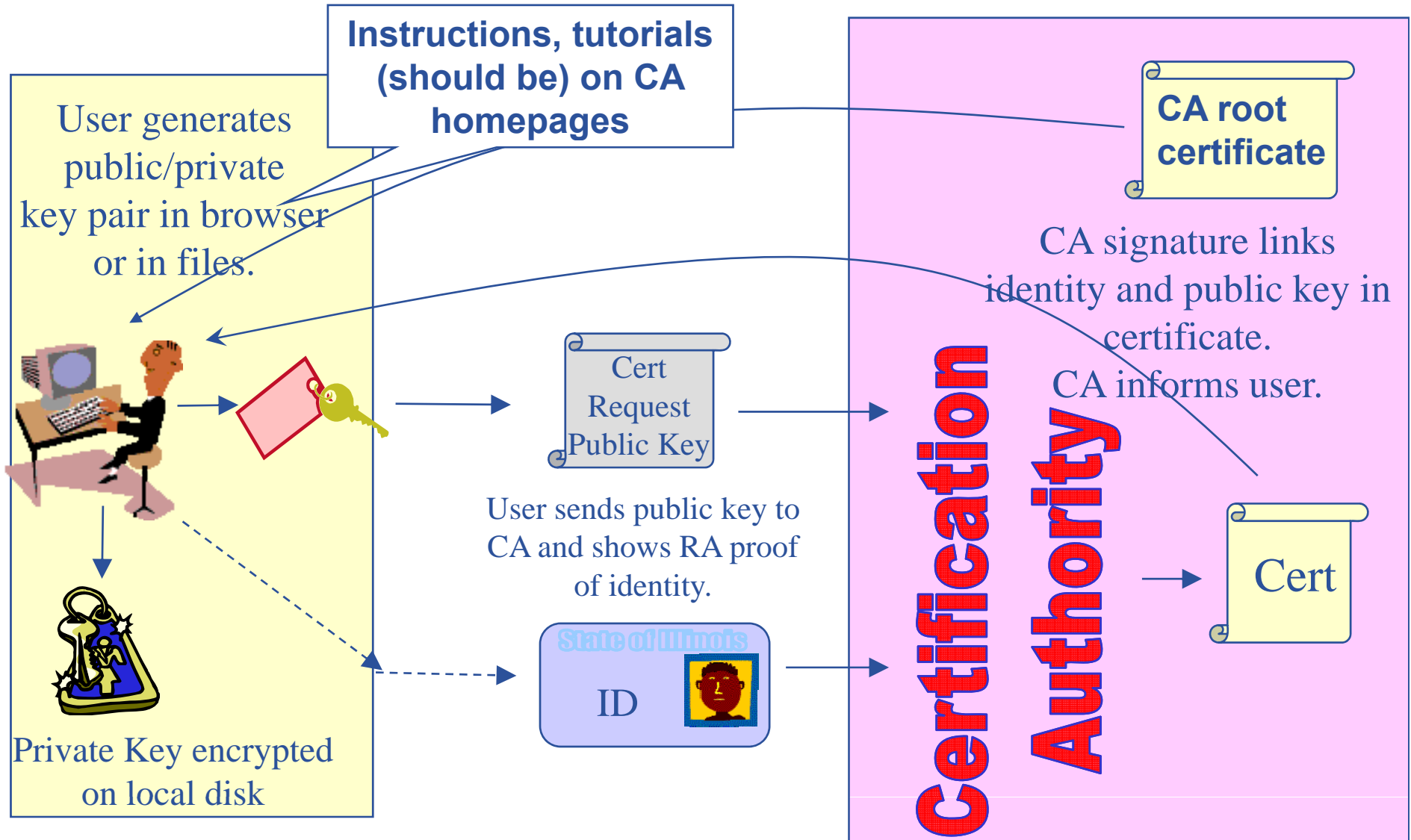
After A and B authenticated each other, for A to send a message to B:

- **Default: message integrity checking**
 - Not private – a test for tampering

- **For private communication:**
 - Encrypt all the message (not just hash) - Slower



Issuing a grid certificate



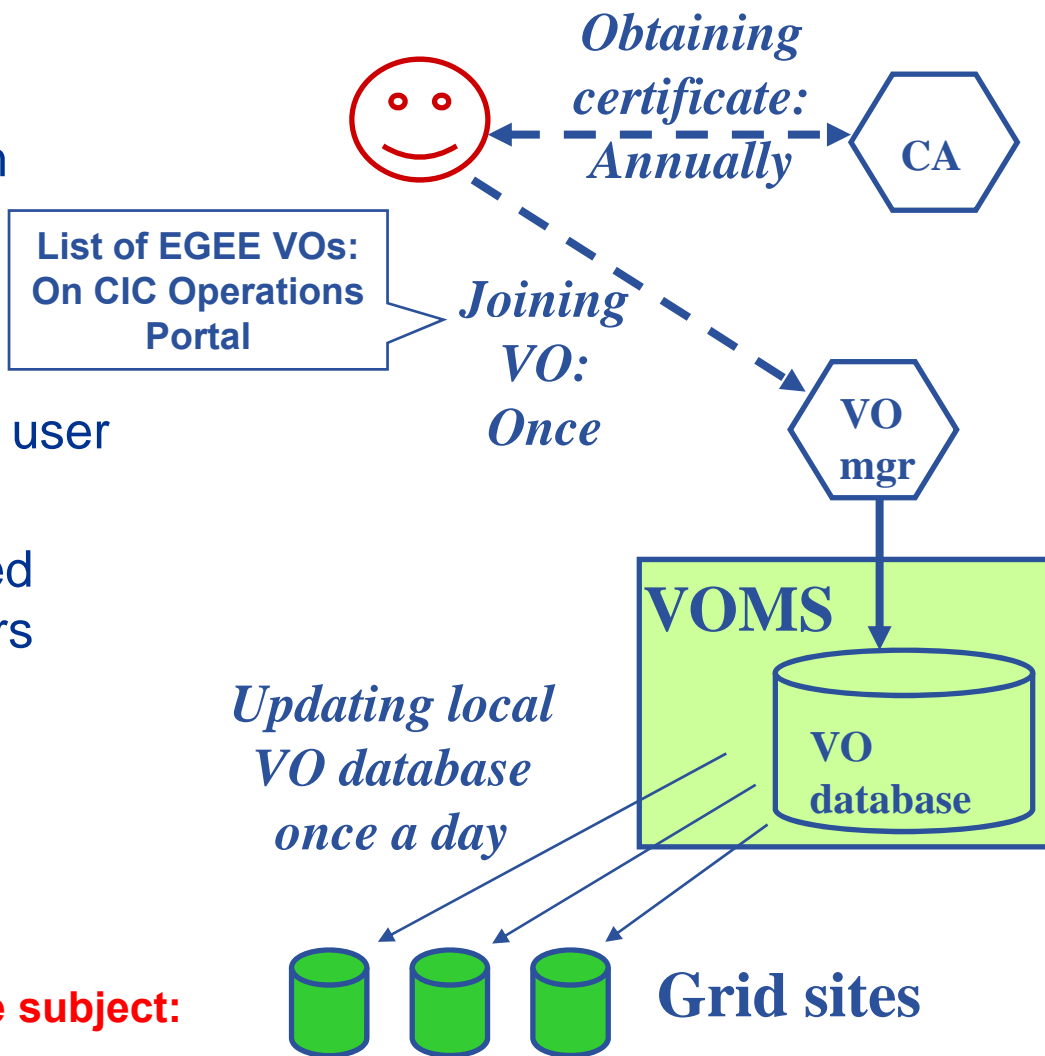
- **Keep your private key secure**
 - if possible *on a USB drive only*
- **Do not loan your certificate to anyone**
- **Report to your local/regional contact if your certificate has been compromised.**
- **Note file access rights:**

```
[sipos@glite-tutor sipos]$ ls -l .globus/
total 8
-rw-r--r--    1 sipos    users    1761 Oct 25  2006 usercert.pem
-r-----    1 sipos    users    951  Oct 24  2006 userkey.pem
```

If your certificate is used by someone other than you, it cannot be proven that it was not you.

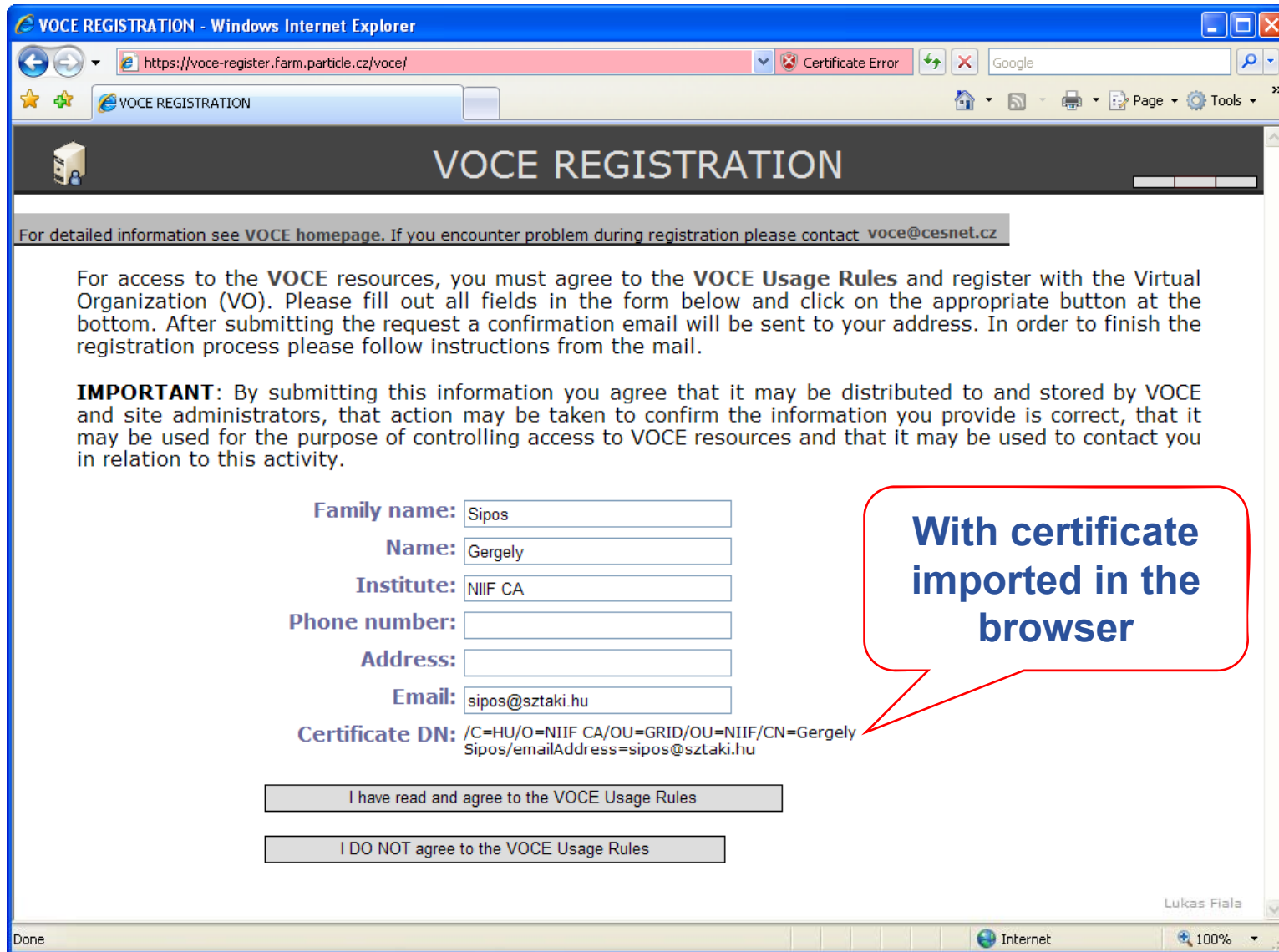
- **Steps**

- User obtains certificate from Certificate Authority
- User registers at the VO
 - usually via a web form
- VO manager authorizes the user
 - VO DB updated
- User information is replicated onto the sites within 24 hours



User's identity in the Grid = certificate subject:

/C=HU/O=NIIF CA/OU=GRID/OU=NIIF/CN=Gergely Sipos/Email=sipos@sztaki.hu



VOCE REGISTRATION - Windows Internet Explorer

https://voce-register.farm.particle.cz/voce/ Certificate Error Google

VOCE REGISTRATION

For detailed information see [VOCE homepage](#). If you encounter problem during registration please contact voce@cesnet.cz

For access to the **VOCE** resources, you must agree to the **VOCE Usage Rules** and register with the Virtual Organization (VO). Please fill out all fields in the form below and click on the appropriate button at the bottom. After submitting the request a confirmation email will be sent to your address. In order to finish the registration process please follow instructions from the mail.

IMPORTANT: By submitting this information you agree that it may be distributed to and stored by VOCE and site administrators, that action may be taken to confirm the information you provide is correct, that it may be used for the purpose of controlling access to VOCE resources and that it may be used to contact you in relation to this activity.

Family name:

Name:

Institute:

Phone number:

Address:

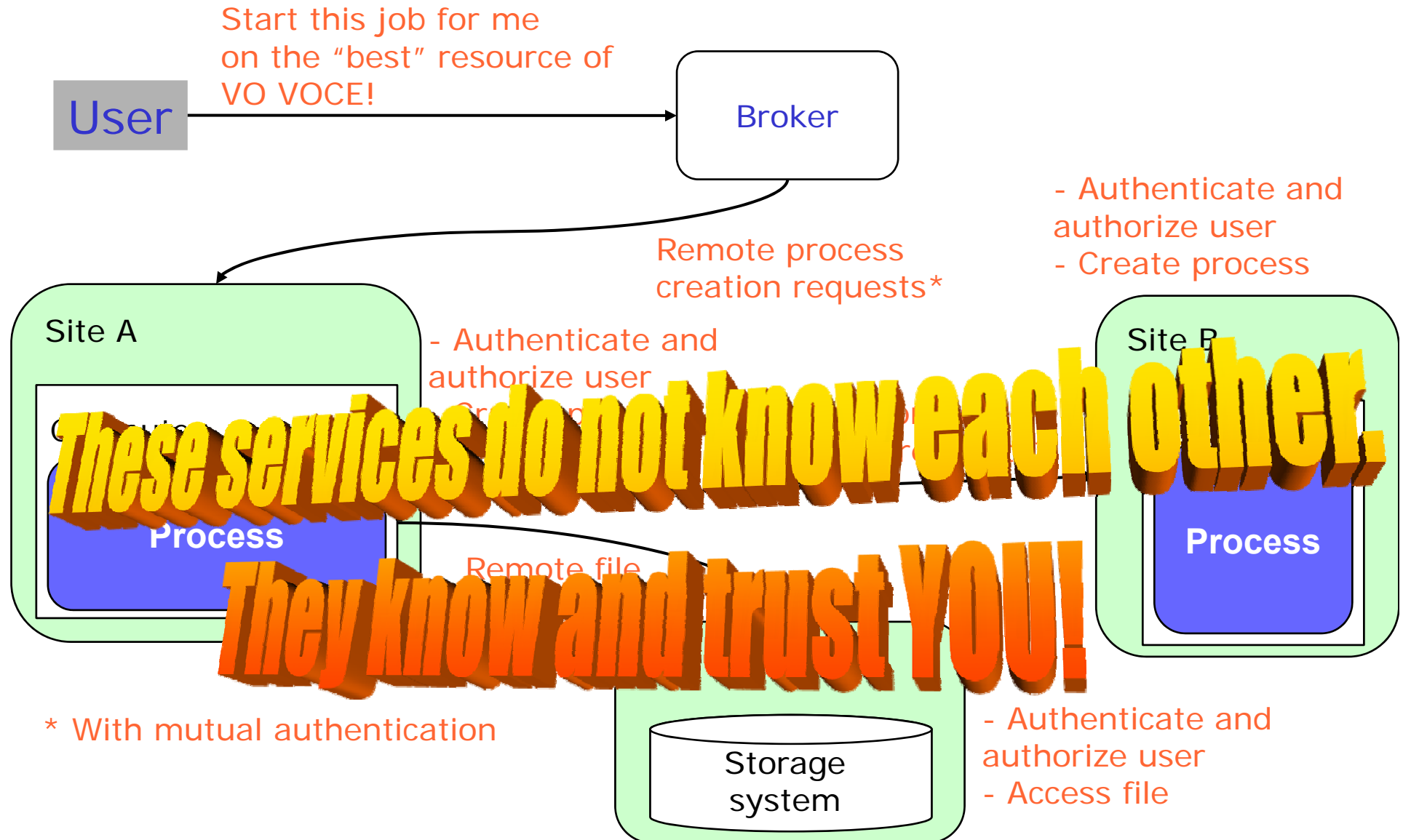
Email:

Certificate DN: /C=HU/O=NIIF CA/OU=GRID/OU=NIIF/CN=Gergely Sipos/emailAddress=sipos@sztaki.hu

Lukas Fiala

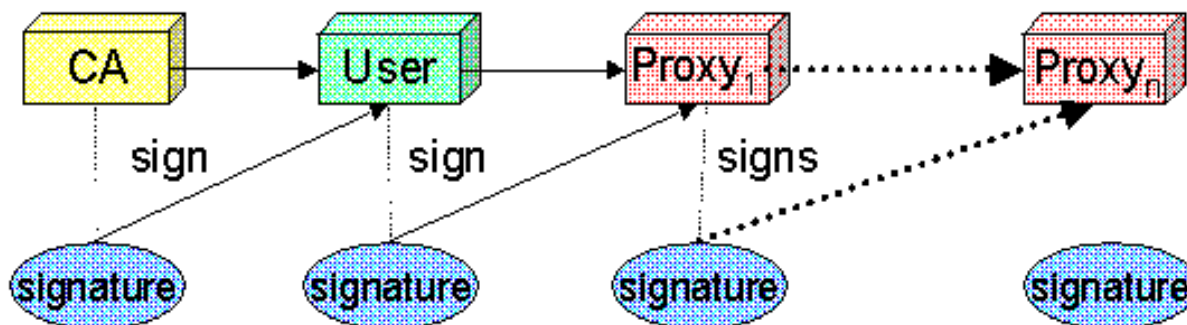
Done Internet 100%

With certificate imported in the browser



* With mutual authentication

- **Delegation** - allows remote process and services to authenticate **on behalf of the user**
 - Remote process/service “**impersonates**” the user
- **Achieved by creation of next-level key-pair from a user key-pair: proxy**
 - Proxy has limited lifetime
 - Proxy may be valid for limited operations
- **The client can delegate the proxy to processes**
 - Each service decides whether it accepts proxies for authentication



- It is created usually by the `voms-proxy-init` command:

```
[sipos@glite-tutor sipos]$ voms-proxy-init --voms gilda
Enter GRID pass phrase: *****
Your identity: /C=HU/O=NIIF CA/OU=GRID/OU=NIIF/CN=Gergely
Sipos/Email=sipos@sztaki.hu
Creating temporary proxy ..... Done
Contacting voms.ct.infn.it:15001 [/C=IT/O=INFN/OU=Host/L=Catania/CN=voms.ct.infn.it]
"gilda" Done
Creating proxy ..... Done
Your proxy is valid until Sat Jun 23 04:55:19 2007
```

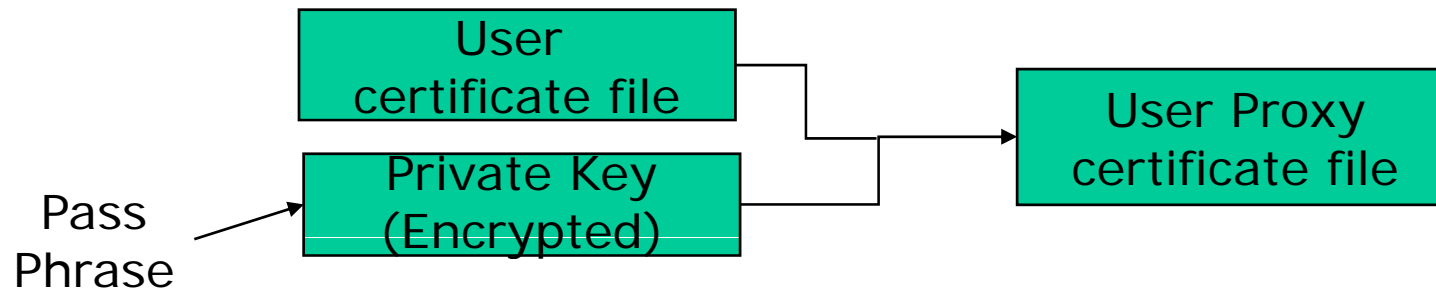
% `voms-proxy-init` → login to the Grid

Enter PEM pass phrase: ***** → private key is protected by a password

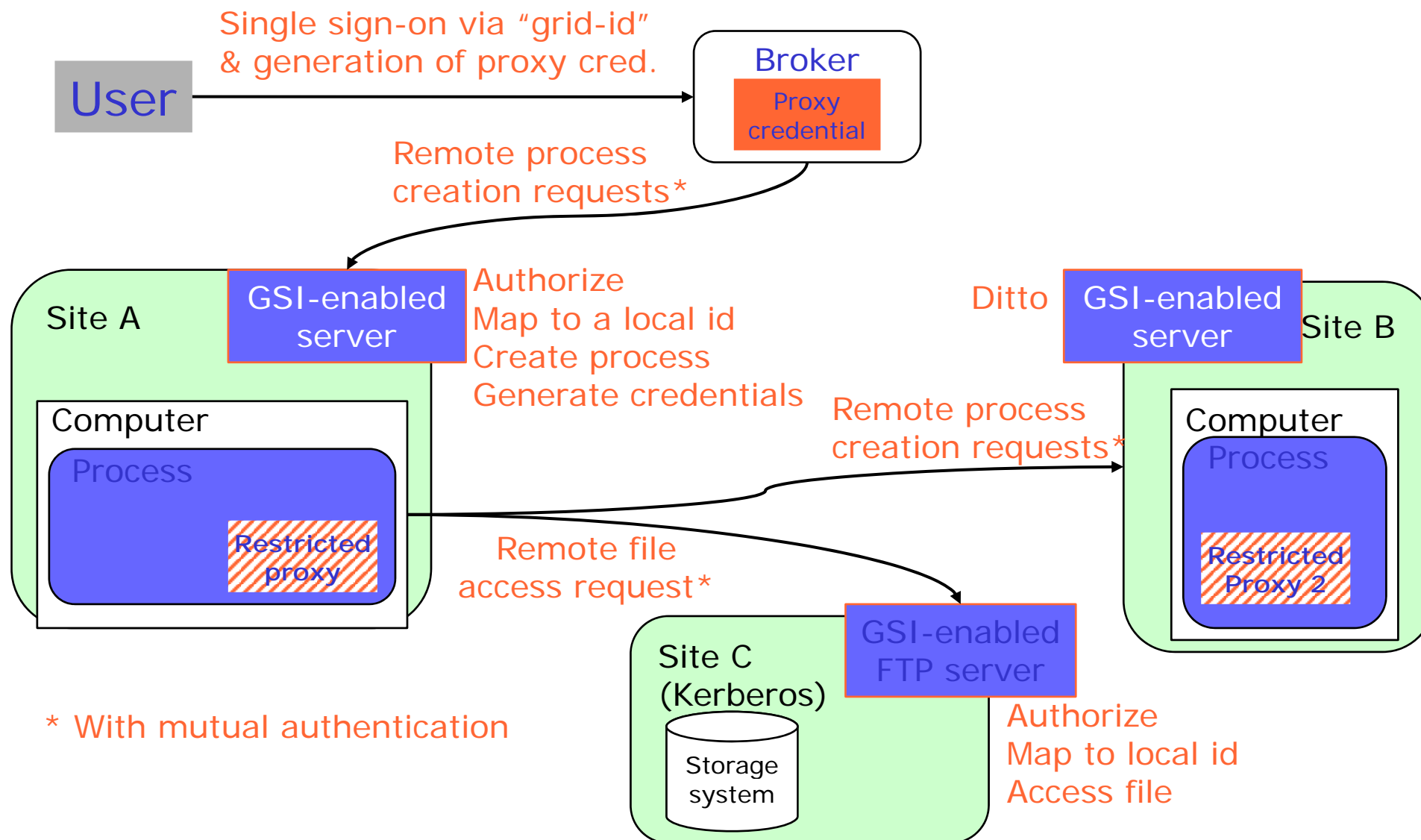
– Options for `voms-proxy-init`:

- VO name
- `-hours` <lifetime of new 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 during proxy creation!

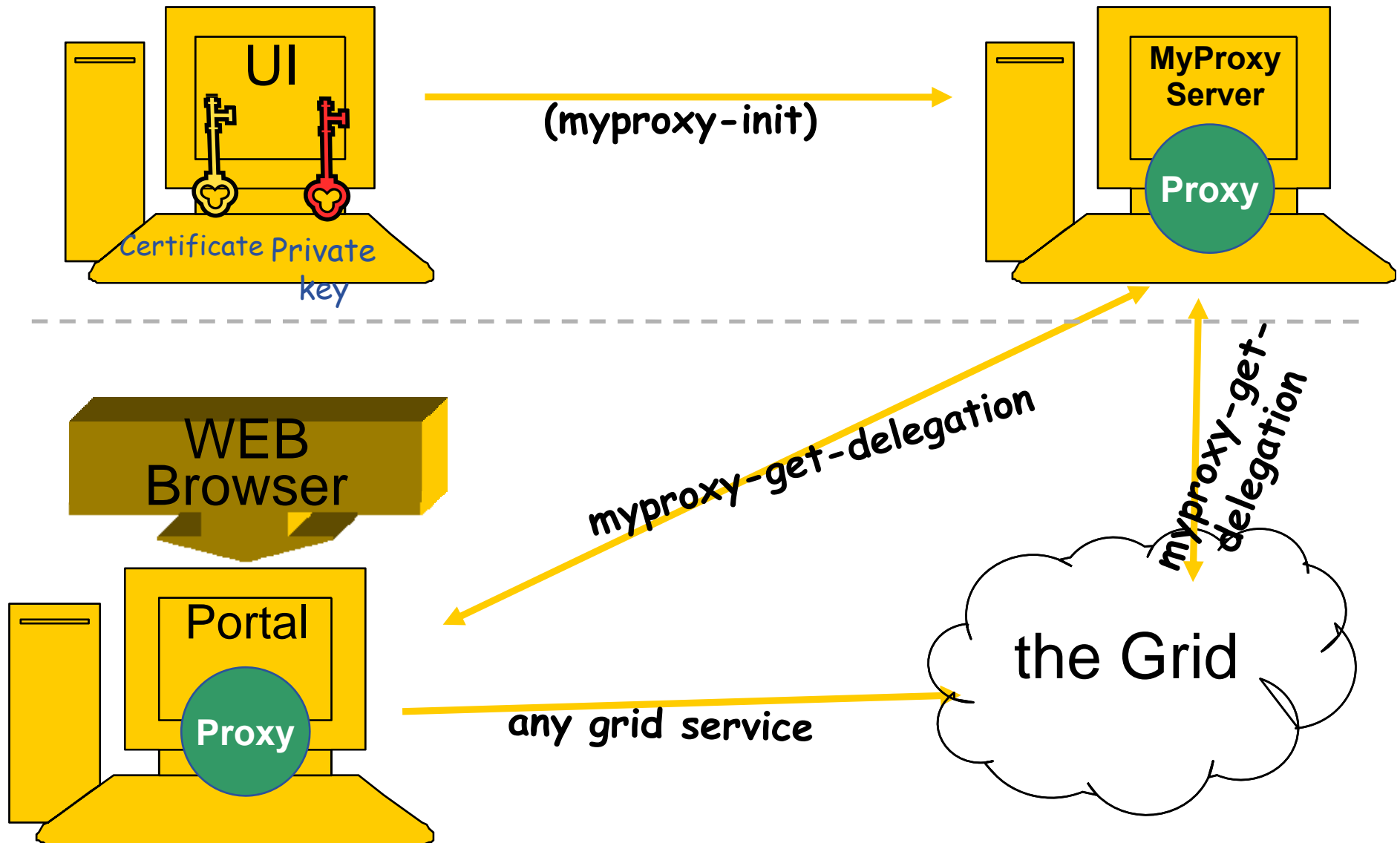


* With mutual authentication

- **voms-proxy-init** \equiv “login to the Grid”
- **To “logout” you have to destroy your proxy:**
 - `voms-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:**
 - `voms-proxy-info`
 - Options for printing proxy information

<code>-subject</code>	<code>-issuer</code>
<code>-type</code>	<code>-timeleft</code>
<code>-strength</code>	<code>-help</code>

- **You may need:**
 - To interact with a grid from many machines
 - And you realise that you must NOT, EVER leave your certificate where anyone can find and use it....
 - To use a portal and delegate to the portal the right to act on your behalf (First step is for the portal to make a proxy certificate for you)
 - To run jobs that might last longer than the lifetime of a short-lived proxy
- **Solution: you can store a proxy in a “MyProxy server” and derive a proxy certificate when needed.**
- **Most often used commands:**
 - myproxy-init -s <host_name>
 - *create and store a long term proxy certificate*
 - myproxy-info
 - get information about stored long living proxy
 - myproxy-get-delegation
 - get a new proxy from the MyProxy server
 - myproxy-destroy
 - Remove the proxy from MyProxy

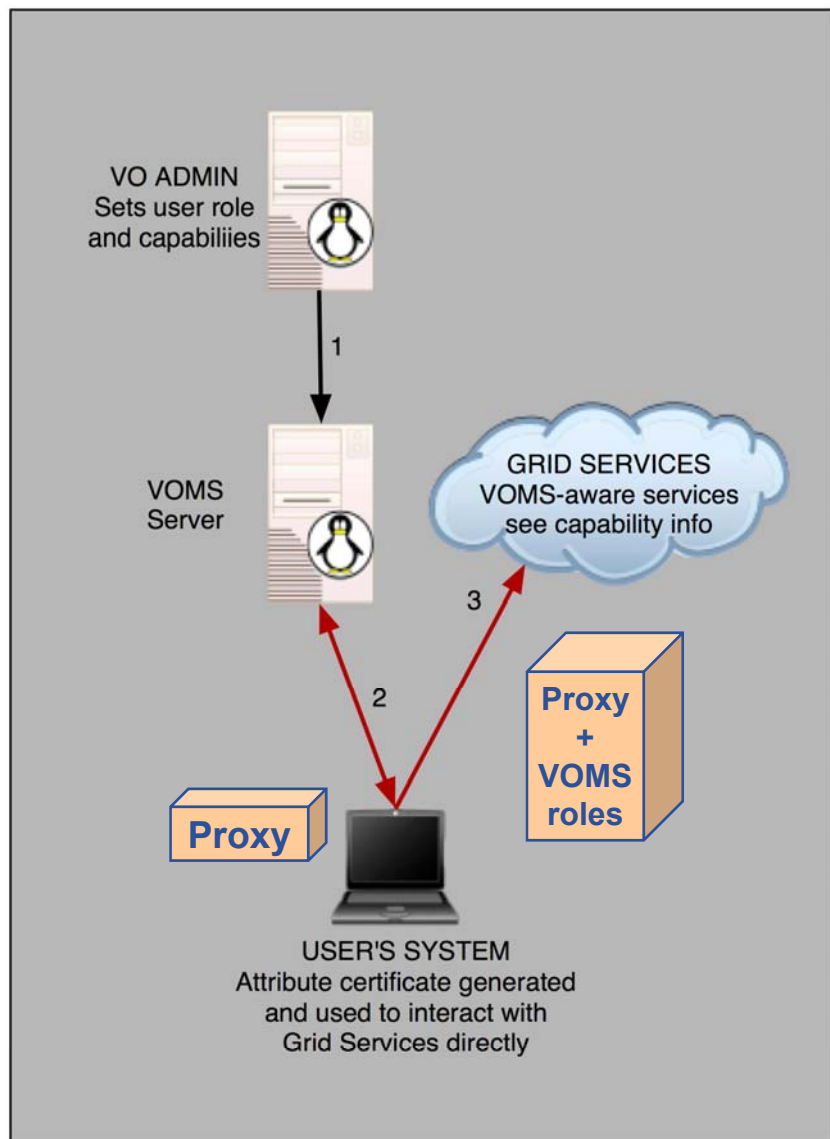


Before VOMS

- All VO members have same rights
- Grid user identities are mapped onto local user accounts statically
- User is authorised as a member of a single VO (no aggregation of roles)
- `grid-proxy-init`

VOMS

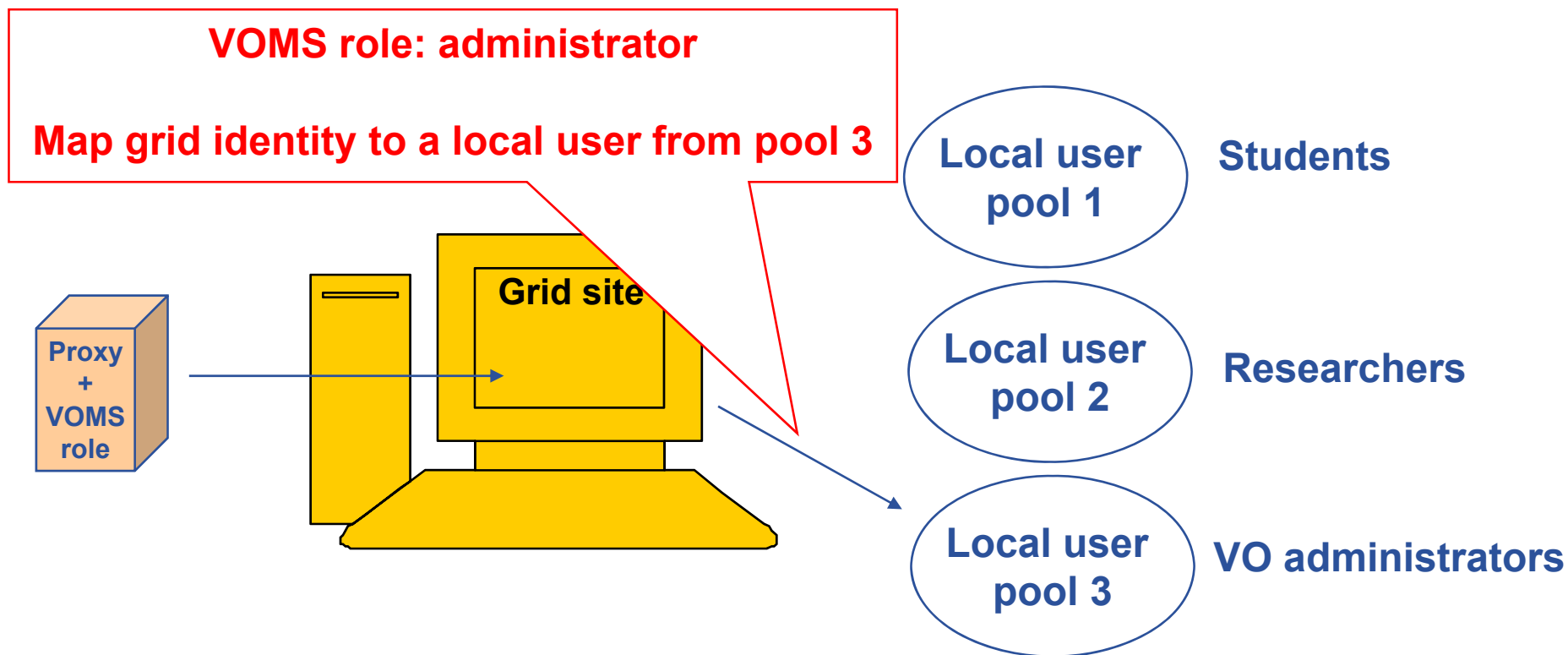
- **VO can have groups**
 - Different rights for each
 - Different groups of experimentalists
 - ...
 - Nested groups
- **VOMS has roles**
 - Assigned to specific purposes
 - E.g. system admin
 - When assume this role
- **User can be in multiple VOs**
 - Aggregate roles
- **Proxy certificate carries the additional attributes**
- `voms-proxy-init`



- **A community-level group membership system**
- **Database of user roles**
 - Administrative tools
 - Client interface
- **voms-proxy-init**
 - Creates a proxy locally
 - Contacts the VOMS server and extends the proxy with a role

`voms-proxy-init -voms voce`

- **Allows VOs to centrally manage user roles**



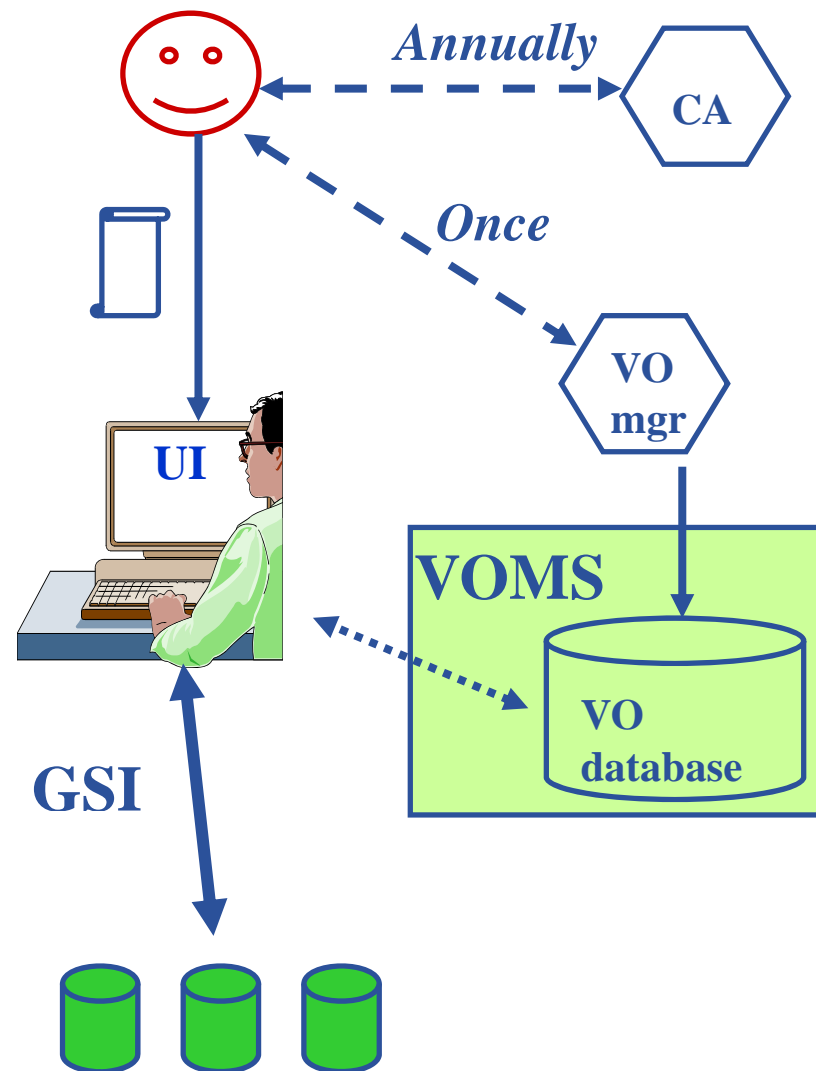
The grid user can perform those actions on the site that any user account from pool 3 is allowed to

- **Authentication**

- User obtains certificate from Certificate Authority
- Connects to UI by ssh and uploads certificate to UI
 - or
 - Login to a portal and use MyProxy
- Single logon to the Grid - create proxy
- then **Grid Security Infrastructure uses proxies**

- **Authorisation**

- User joins Virtual Organisation
- VO manager updates VOMS DB
- Capabilities added to proxy by VOMS



- Do not launch a delegation service for longer than your current task needs.

If your certificate *or delegated service* is used by someone other than you, it cannot be proven that it was not you.



Enabling Grids for E-scienceE

Thank you!

Questions?

www.eu-egEE.org

