omii europe

open middleware infrastructure institute

Middleware Basics - 1

Mike Mineter
NeSC-TOE
mjm@nesc.ac.uk

Contents

- **Basic components typical of grids**
- **Importance of Authorisation and Authentication**
  - Getting and using a certificate
- **Virtual organisations**

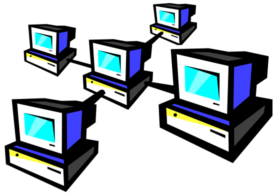
Acknowledgement: slides taken from EGEE training courses

Main components

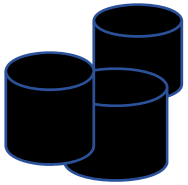
Access service How users logon to a Grid

Resource Broker (RB): Service that matches the user’s requirements with the available resources on a Grid

Information System: Characteristics and status of resources



Computing Element (CE): A batch queue on a site’s computers where the user’s job is executed



Storage Element (SE): provides (large-scale) storage for files

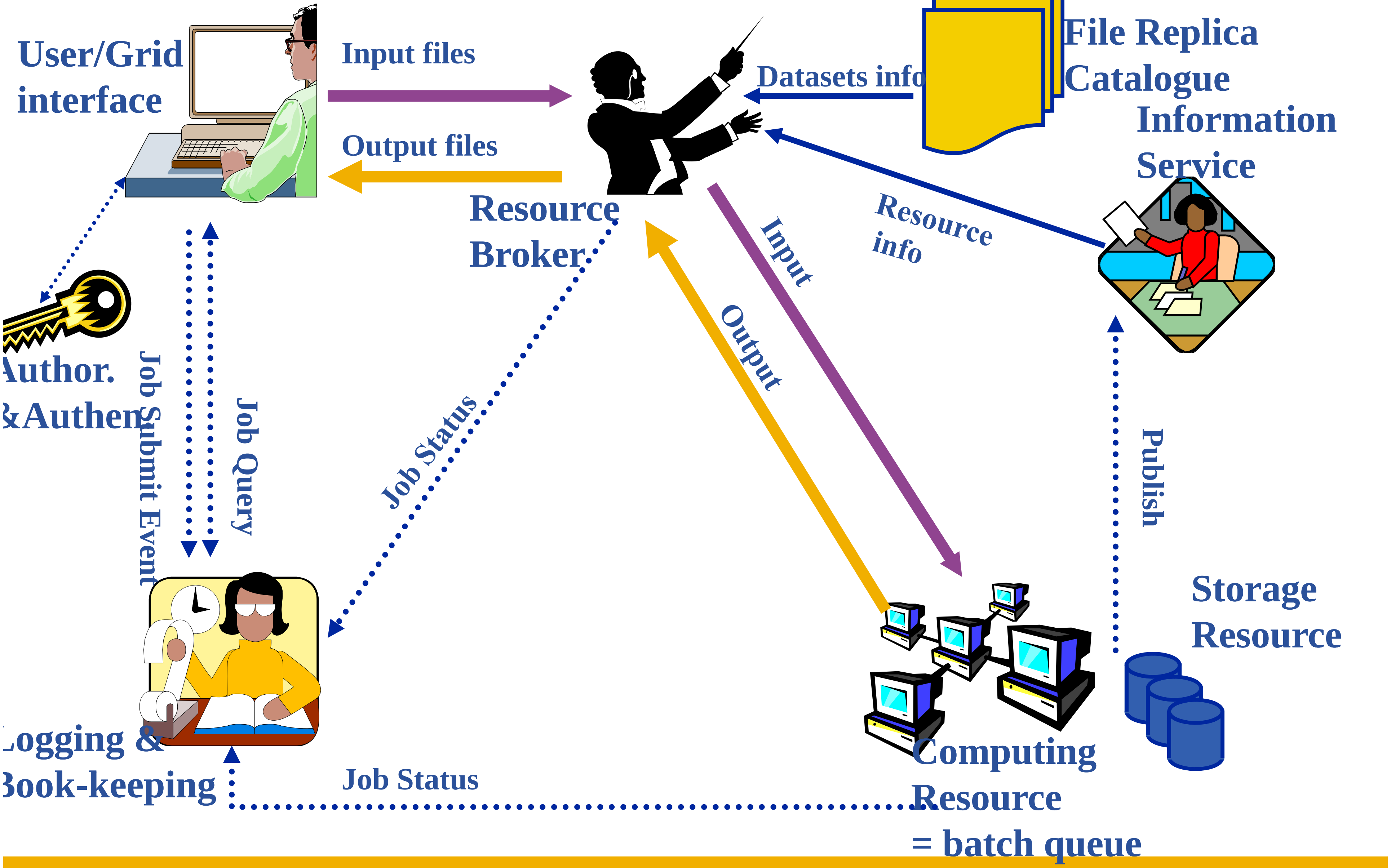
Who provides the resources?! (on EGEE)

Service	**Provider**	**Note**
Access service	**User / institute/ VO / grid operations**	**Computer with client software**
Resource Broker (RB)	**VO / grid operations**	
***Information System*:**	**Grid operations**	
Computing Element (CE)	**VO / sometimes centralised provision also**	**Scalability requires that VOs provide resources to match average need**
Storage Element (SE)	**ditto**	**ditto**

"VO": virtual organisation

"Grid operations": funded effort

Current EGEE grid



2 common types of data services

- **Simple data files on grid-specific storage**
- **Middleware supporting**
  - **Replica files**
    - to be close to where you want computation
    - For resilience
  - **Logical filenames**
  - **Catalogue**: maps logical name to physical storage device/file
  - **Virtual filesystems,** POSIX-like I/O
  - Services provided: storage, transfer, catalogue that maps logical filenames to replicas.
- **Solutions include**
  - **gLite data service**
  - **Globus: Data Replication Service**
  - **Storage Resource Broker**

- **Other data!   e.g. ….**
  - Structured data: RDBMS, XML databases,…
  - Files on project's filesystems
  - Data that may already have other user communities not via Grid
- Require extendable middleware tools to support
  - Computation near to data
  - Controlled exposure of data *without replication*
- Basis for integration and federation
- **OGSA –DAI**
  - In Globus 4
  - Not (yet...) in gLite

Contents

- Basic components typical of grids

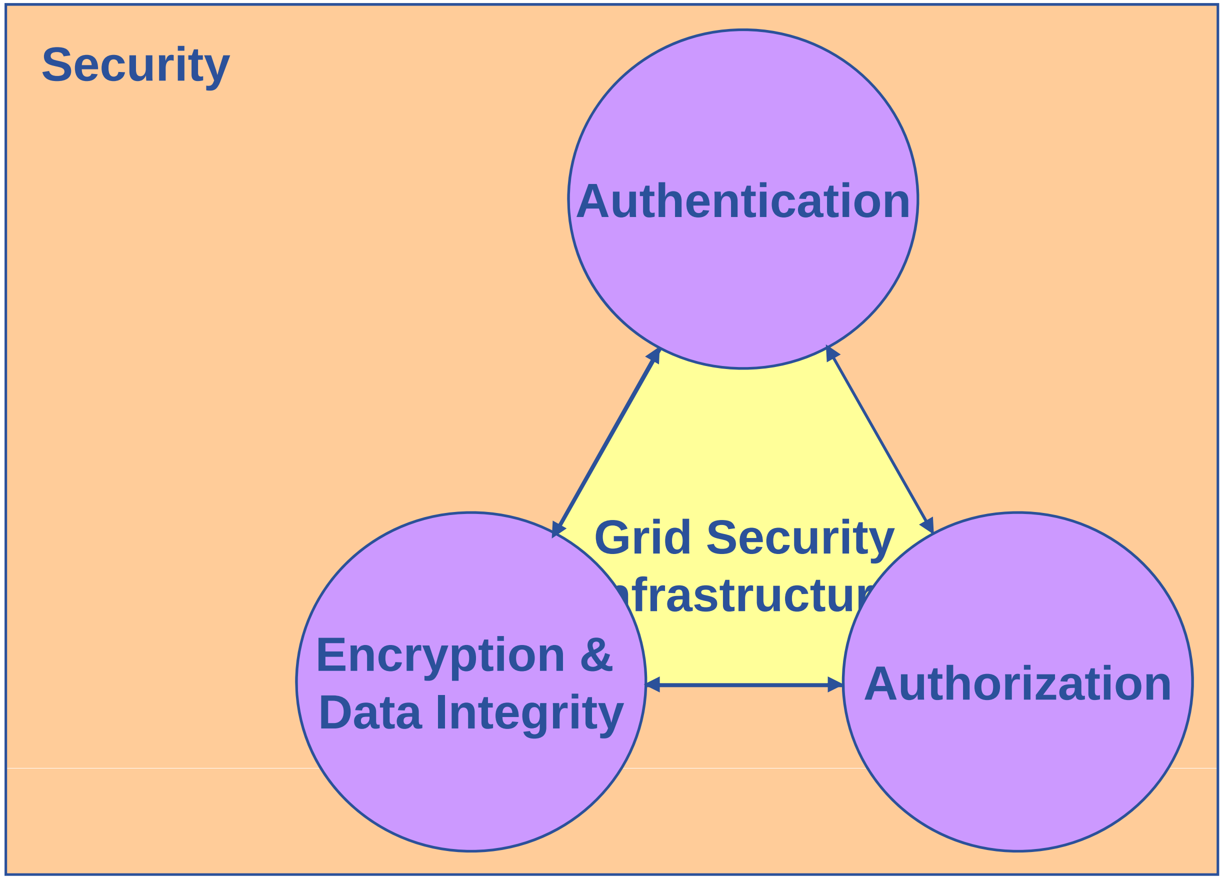
- **Importance of Authorisation and Authentication**
  - Getting and using a certificate

- **Virtual Organisations**

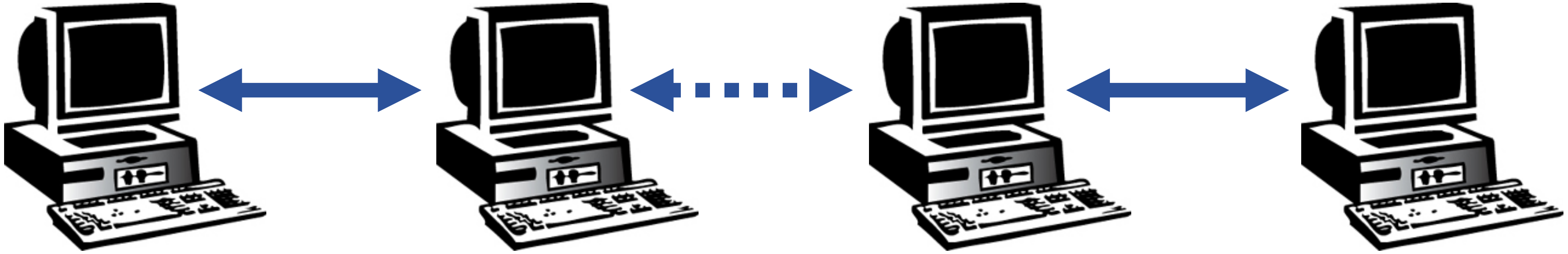
Grid security and trust

- **Providers of resources (computers, databases,..) need risks to be controlled: they are asked to trust users they do not know**
- **User's need**
  - single sign-on: to be able to logon to a machine that can pass the user's identity to other resources
  - To trust owners of the resources they are using
- **Build middleware on layer providing:**
  - *Authentication:* know who wants to use resource
  - *Authorisation:* know what the user is allowed to do
  - *Security:* reduce vulnerability, e.g. from outside the firewall
  - *Non-repudiation:* ~ knowing who did what
- **The "Grid Security Infrastructure" middleware is the basis of (most) production grids (EGEE and NGS)**

Security Overview



The Problems - 1



User

Resource

- How does a user securely access the Resource without having any contact with resource owner?

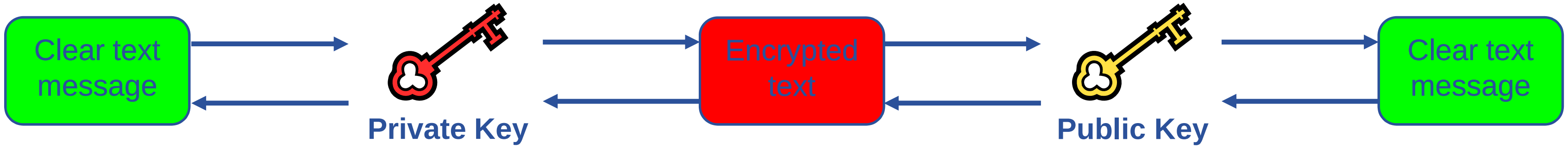
- How does the Resource know who a user is?
- How are authorisation decisions made (resources remain under control of their owner)

The Problems -2: Reducing vulnerability

- **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**
- **Damage caused by viruses, worms etc.**
  - Highly connected infrastructure means worms spread faster than on the internet in general.

Basis of security & authentication

- **Asymmetric encryption…**

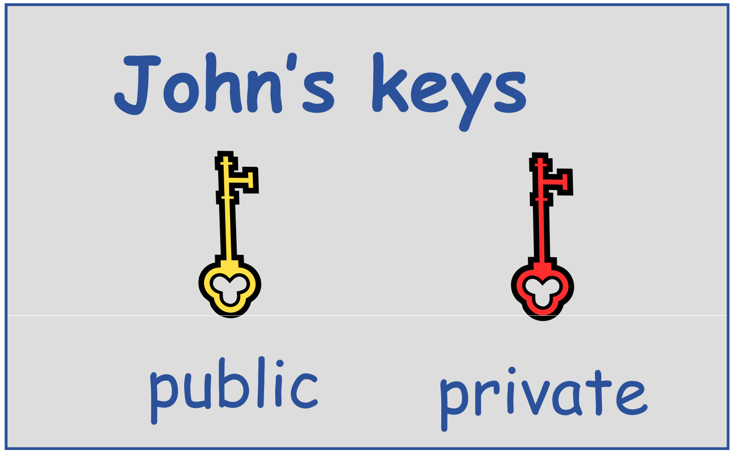


- **…. and Digital signatures …**
  - A hash derived from the message and encrypted with the signer’s private key
  - Signature is checked by decrypting with the signer’s public key

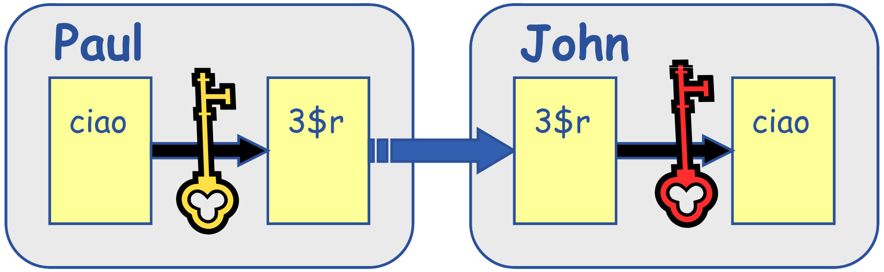
- **Are used to build trust**
  - That a user / site is who they say they are
  - And can be trusted to act in accord with agreed policies

Public Key Algorithms

- **Every user 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.

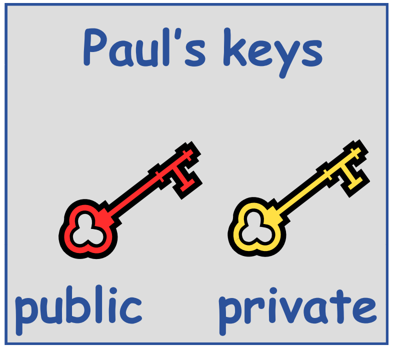


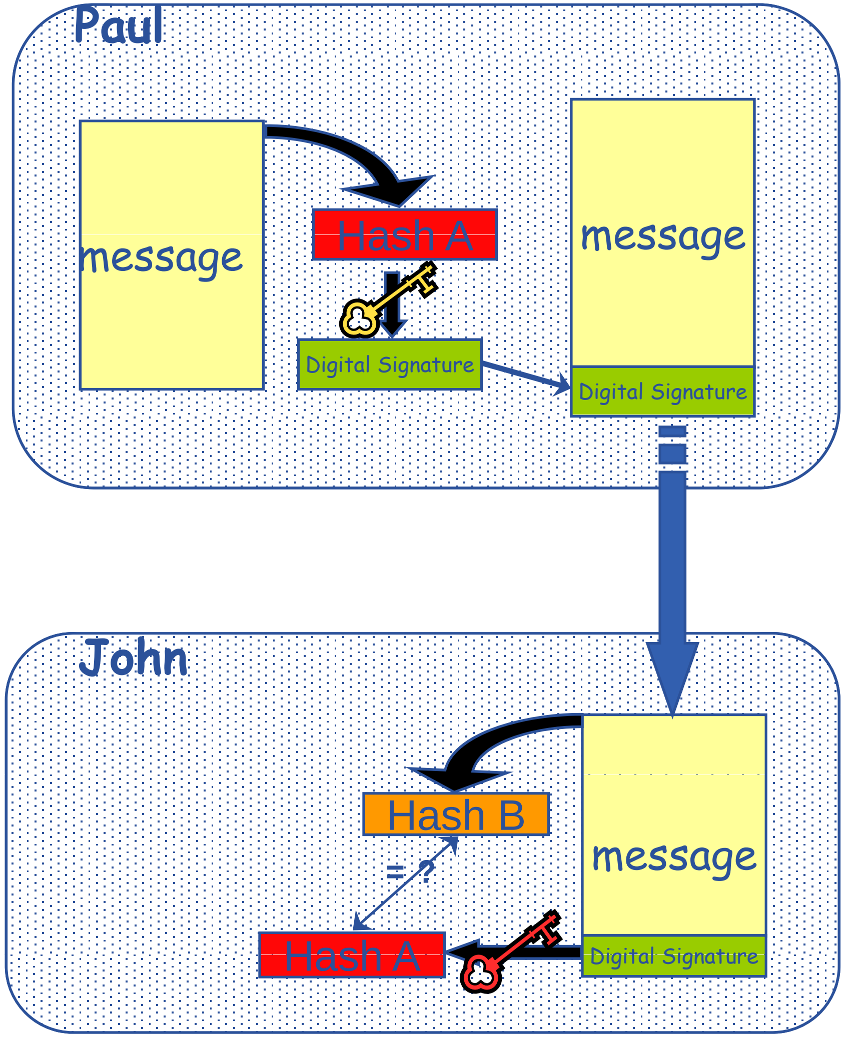
- **Concept - simplified version:**
  - Public keys are exchanged
  - The sender encrypts using receiver's public key
  - The receiver decrypts using their private key;



Digital Signature

- Paul **calculates the** *hash* **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**
- **Decrypts signature, to get A, using Paul's** *public* **key.**
- **If hashes equal:**
  1. message wasn't modified;
  2. hash A is from Paul's private key





Digital Certificates

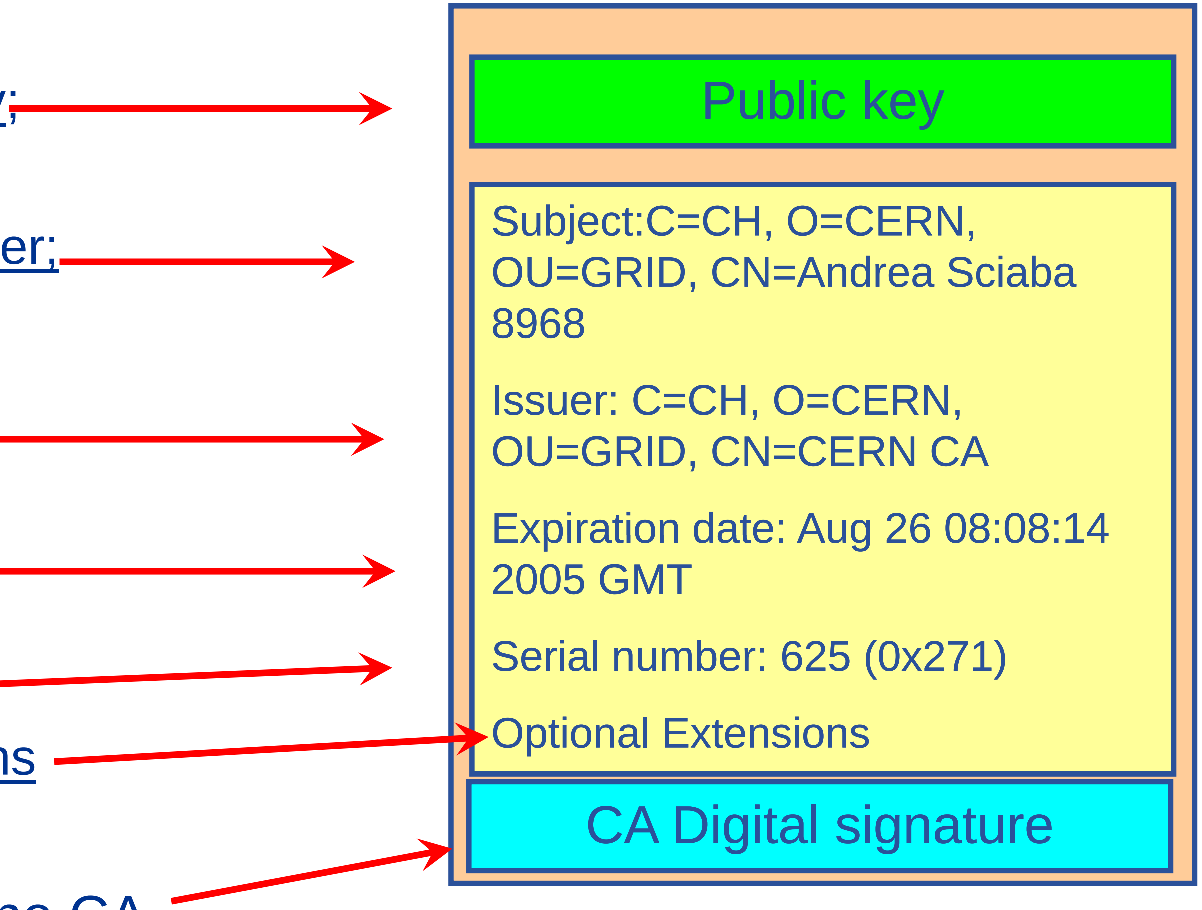
- **How can John be sure that Paul’s public key is really Paul’s public key and not someone else’s?**
  - A *third party* certifies correspondence between the public key and Paul’s identity.
  - Both John and Paul trust this third party

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

X.509 Certificates

- **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



Public key

Subject:C=CH, O=CERN, OU=GRID, CN=Andrea Sciaba 8968

Issuer: C=CH, O=CERN, OU=GRID, CN=CERN CA

Expiration date: Aug 26 08:08:14 2005 GMT

Serial number: 625 (0x271)

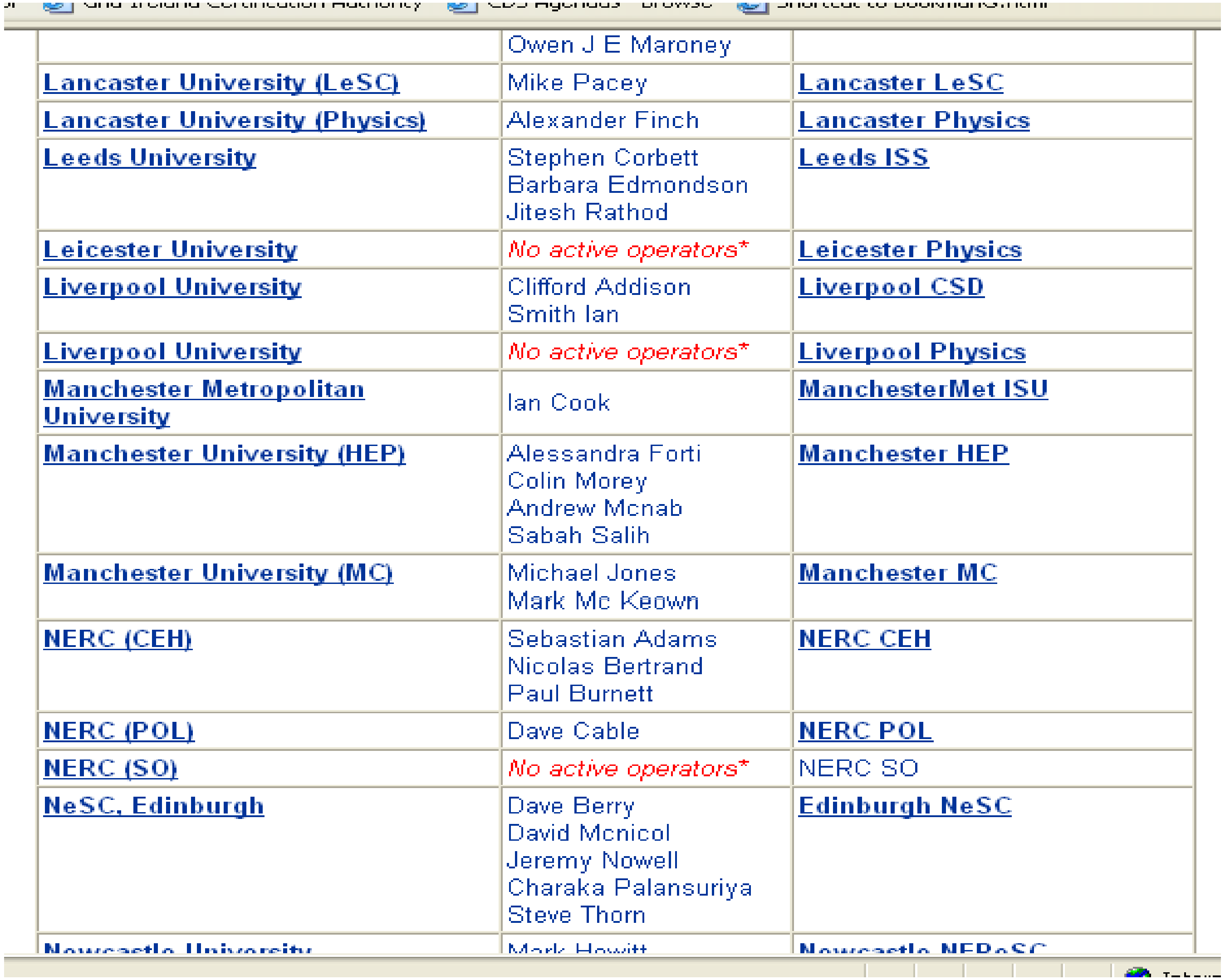
Optional Extensions

CA Digital signature

Certification Authorities

- 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 each establish a number of people "registration authorities" RAs
- To find RAs in UK go to http://www.grid-support.ac.uk/ca/ralist.htm

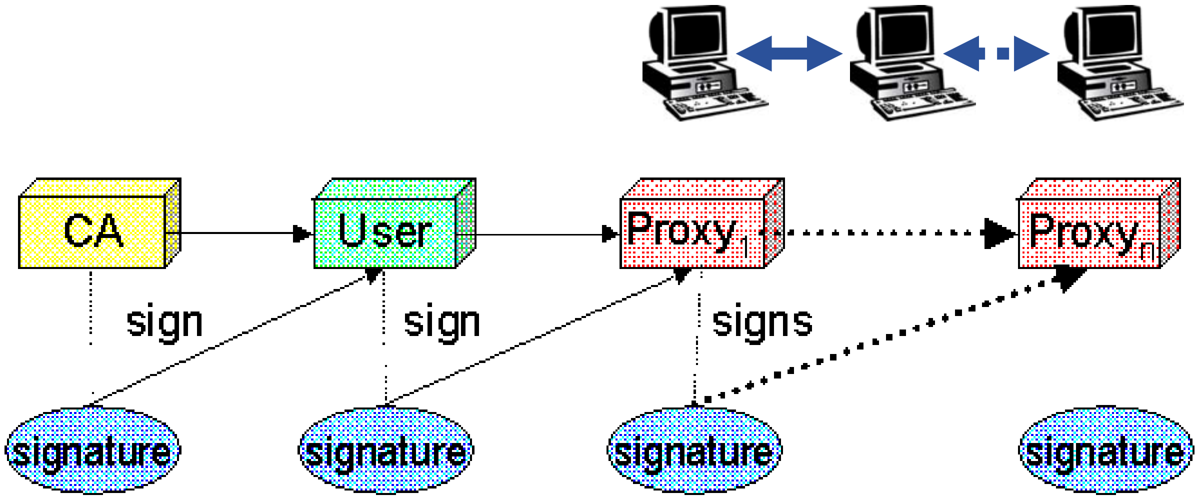
RA list


Grid Ireland Certification Authority
CDS Agendas - Browse
Shortcut to Bookmarks.html

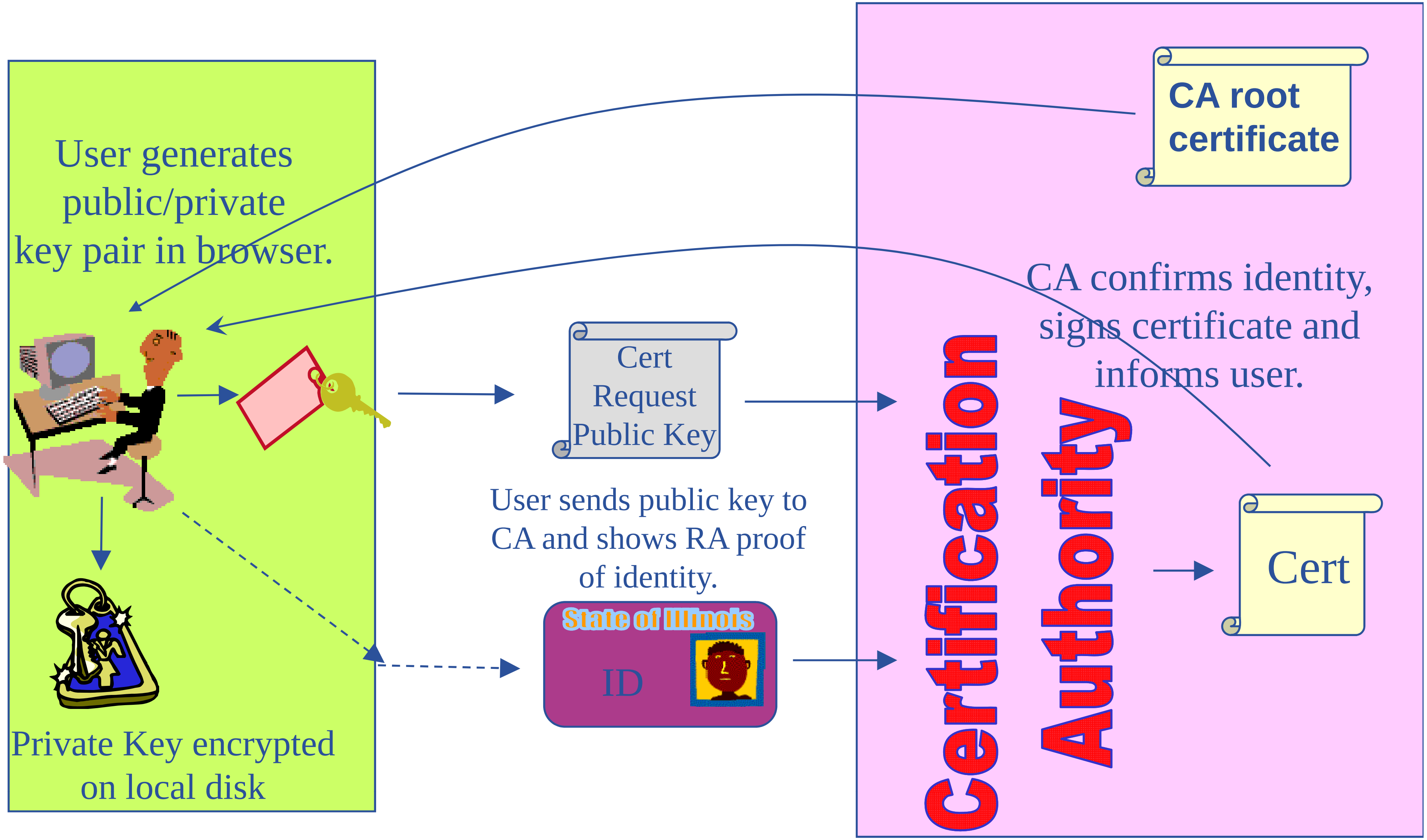
	Owen J E Maroney	
Lancaster University (LeSC)	Mike Pacey	Lancaster LeSC
Lancaster University (Physics)	Alexander Finch	Lancaster Physics
Leeds University	Stephen Corbett Barbara Edmondson Jitesh Rathod	Leeds ISS
Leicester University	No active operators*	Leicester Physics
Liverpool University	Clifford Addison Smith Ian	Liverpool CSD
Liverpool University	No active operators*	Liverpool Physics
Manchester Metropolitan University	Ian Cook	ManchesterMet ISU
Manchester University (HEP)	Alessandra Forti Colin Morey Andrew Mcnab Sabah Salih	Manchester HEP
Manchester University (MC)	Michael Jones Mark Mc Keown	Manchester MC
NERC (CEH)	Sebastian Adams Nicolas Bertrand Paul Burnett	NERC CEH
NERC (POL)	Dave Cable	NERC POL
NERC (SO)	No active operators*	NERC SO
NeSC, Edinburgh	Dave Berry David Mcnicol Jeremy Nowell Charaka Palansuriya Steve Thorn	Edinburgh NeSC
Newcastle University	Mark Hewitt	Newcastle NEReSC

Grid Security Infrastructure - proxies

- **To support delegation: A delegates to B the right to act on behalf of A**
- **proxy certificates *extend X.509 certificates***
  - Short-lived certificates signed by the user's certificate or a proxy
  - Reduces security risk, enables delegation



Certificate Request



User generates public/private key pair in browser.

Private Key encrypted on local disk

User sends public key to CA and shows RA proof of identity.

CA confirms identity, signs certificate and informs user.

User Responsibilities

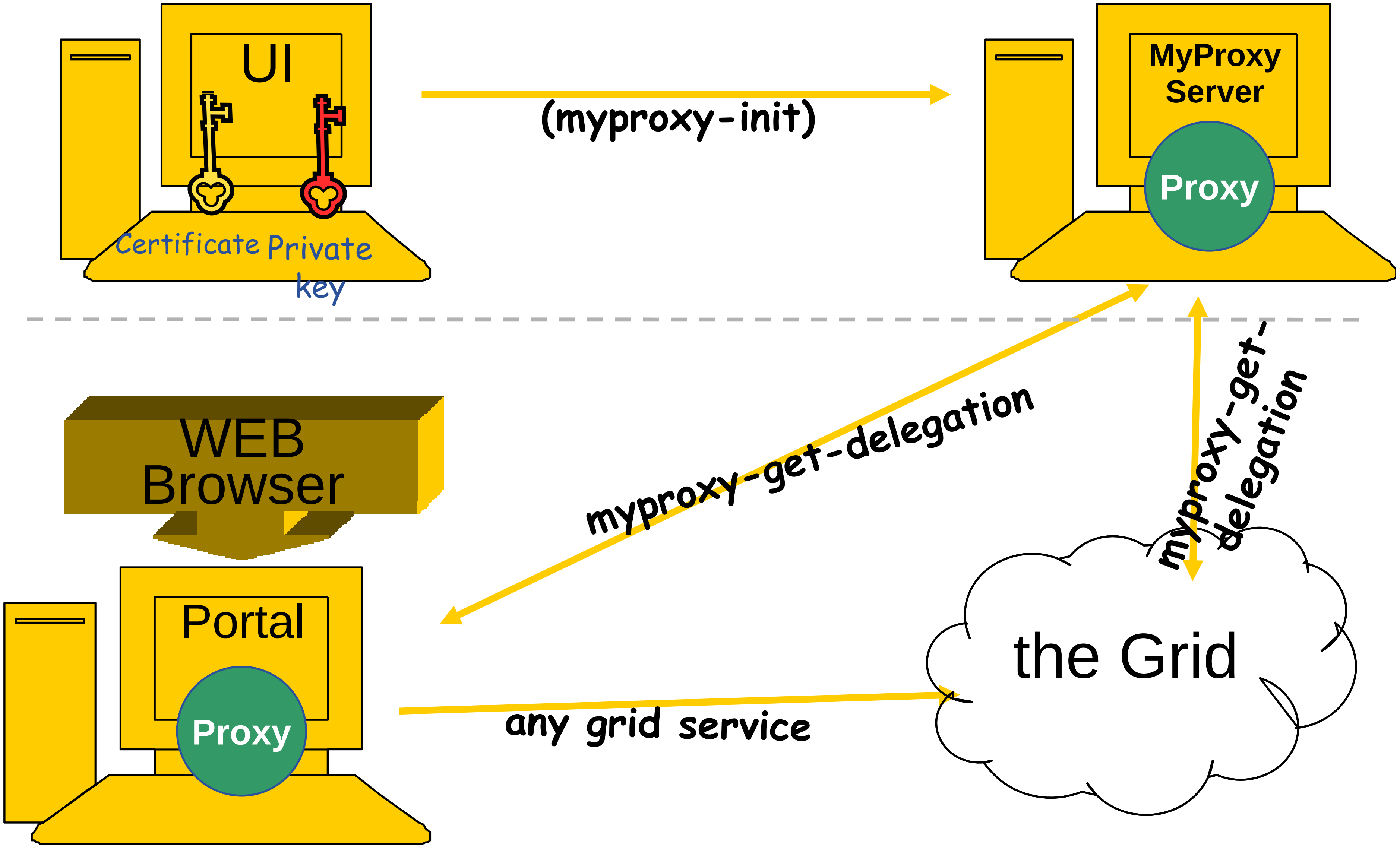
- Keep your private key secure.
- Do not loan your certificate to anyone.
- Report to your local/regional contact if your certificate has been compromised.
- Do not launch a delegation service for longer than your current task needs.

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

MyProxy server

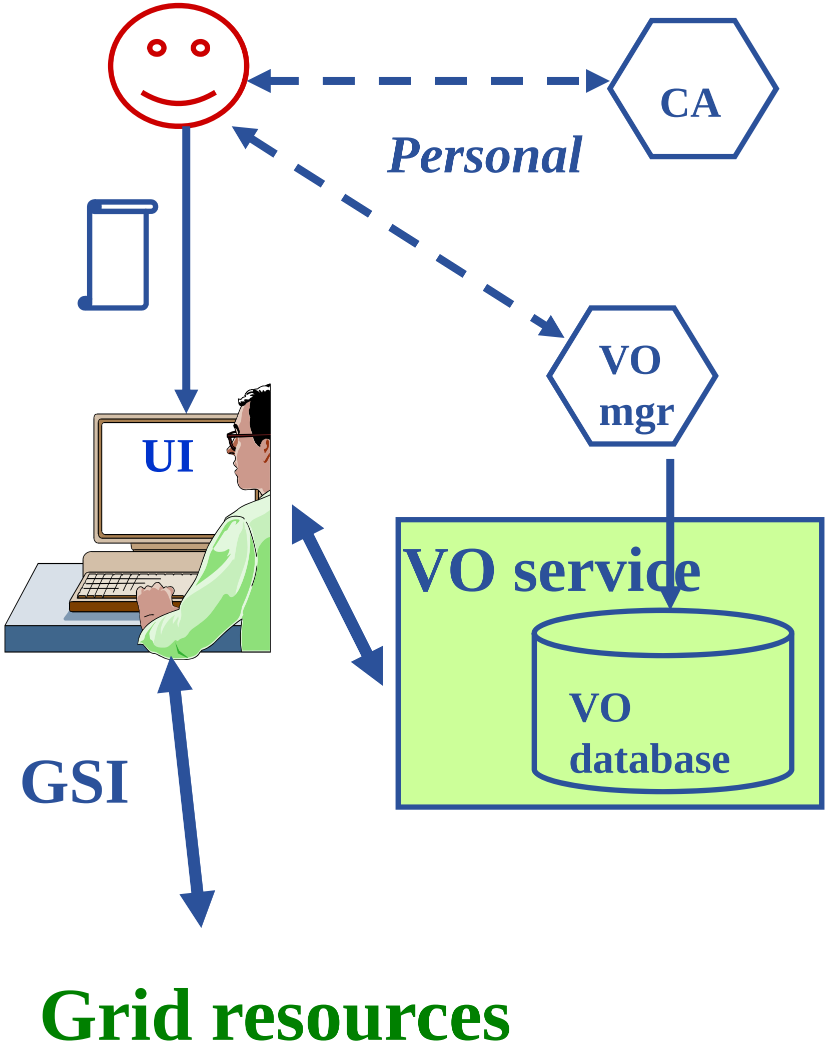
- **You may need:**
  - To interact with a grid from many machines
    - And you realise that you must NOT, EVER leave your certificate where anyone can find and use it….
  - To use a portal and delegate to the portal the right to act on your behalf (First step is for the portal to make a proxy certificate 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

MyProxy examples



Certificates - Summary

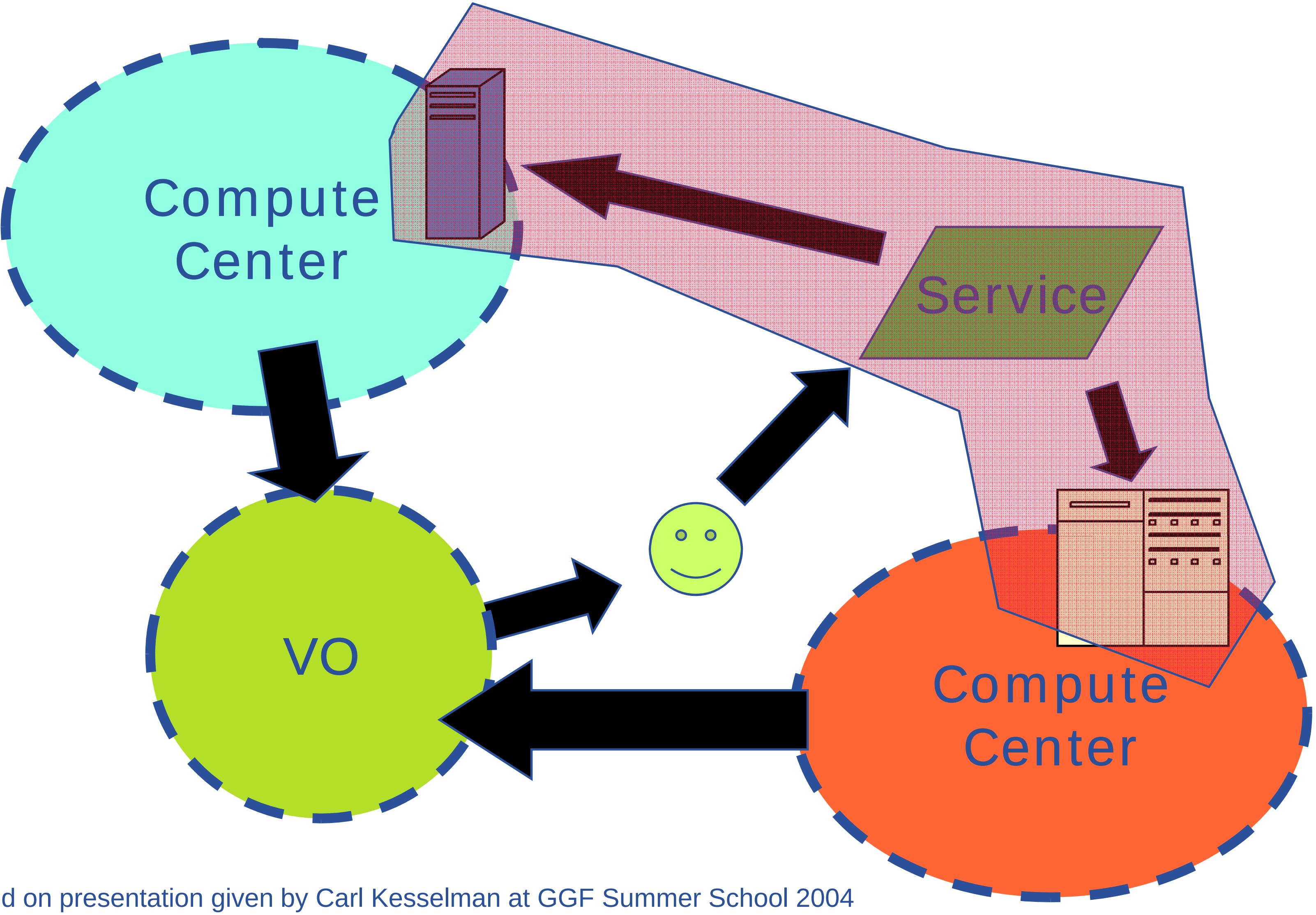
- Authentication
  - User obtains certificate from Certificate Authority
  - Connects to UI by ssh
  - Downloads certificate
  - Single logon – to UI - create proxy
  - then **Grid Security Infrastructure uses proxies to identify users to other machines**
- Authorisation
  - User joins Virtual Organisation
  - VO negotiates access to Grid resources
  - Authorisation tested on receipt of credentials:



Contents

- Basic components typical of grids
- **Importance of Authorisation and Authentication**
  - Getting and using a certificate
- **Virtual Organisations**

The Role of the Virtual Organisation (VO)



slide based on presentation given by Carl Kesselman at GGF Summer School 2004

Controlling user rights: Virtual Organization Membership Service

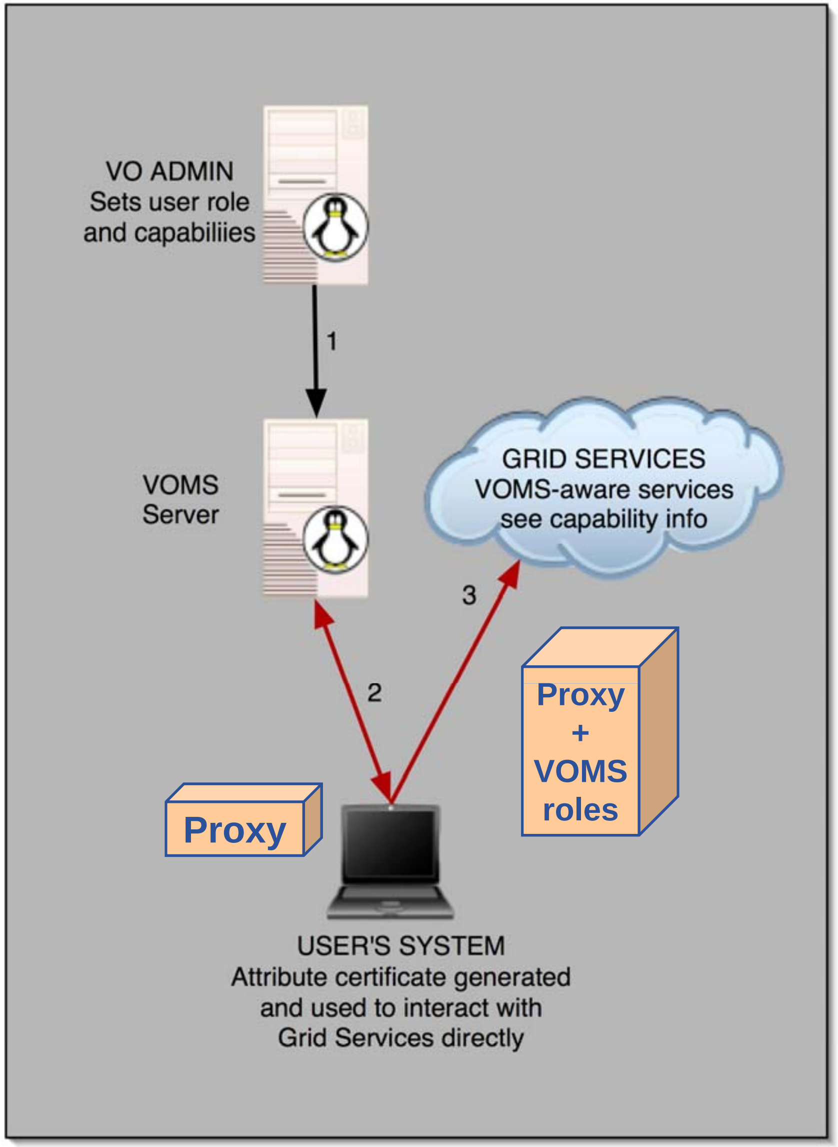
Before VOMS

- **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**

voms-proxy-init in the background



- **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**

VOMS

- VOMS is a grid attribute system that allows a client to embed an attribute certificate in a well known certificate extension. Since the embedded attribute certificate is signed by a VOMS server, a VOMS enabled service can parse and verify this extra certificate and treat the data therein as extra information about the client to use in an authorization decision
- At a glance
  - A VOMS server, typically one for each VO, contains information about a user
  - The VOMS server, when requested, will digitally sign an assertion stating that a particular DN has some particular attributes
  - A client may embed this in its own proxy certificate to "push" it to the service when accessing resources
  - The service, trusting a particular set of VOMS servers for attribute information, can use the attributes to make authorization decisions
- Using a distributed attribute system relieves services of needing to know every detail about the connecting clients.

Summary

- **Basic components typical of grids:**
  - Information service
  - job execution
  - data storage,management, transfer
  - Logging of activity
  - *Application layer built on these – not everyone needs to see these!*

- **Importance of Authorisation and Authentication**
  - Basis of trust
  - Guard your private key!
  - Delegation
  - MyProxy – on your behalf can hold long lived proxy and issue short-lived proxies to you or services such as portals
  - *Application layer builds on these – not everyone needs to see these!*

- **Virtual organisations**
  - VOMS – used by NGS and EGEE
  - Manages membership of VOs:
  - allows groups, roles to be used for authorsiation decisions