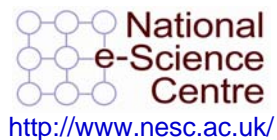# Security, Authorisation and Authentication

Mike Mineter, Guy Warner
Training, Outreach and Education
National e-Science Centre
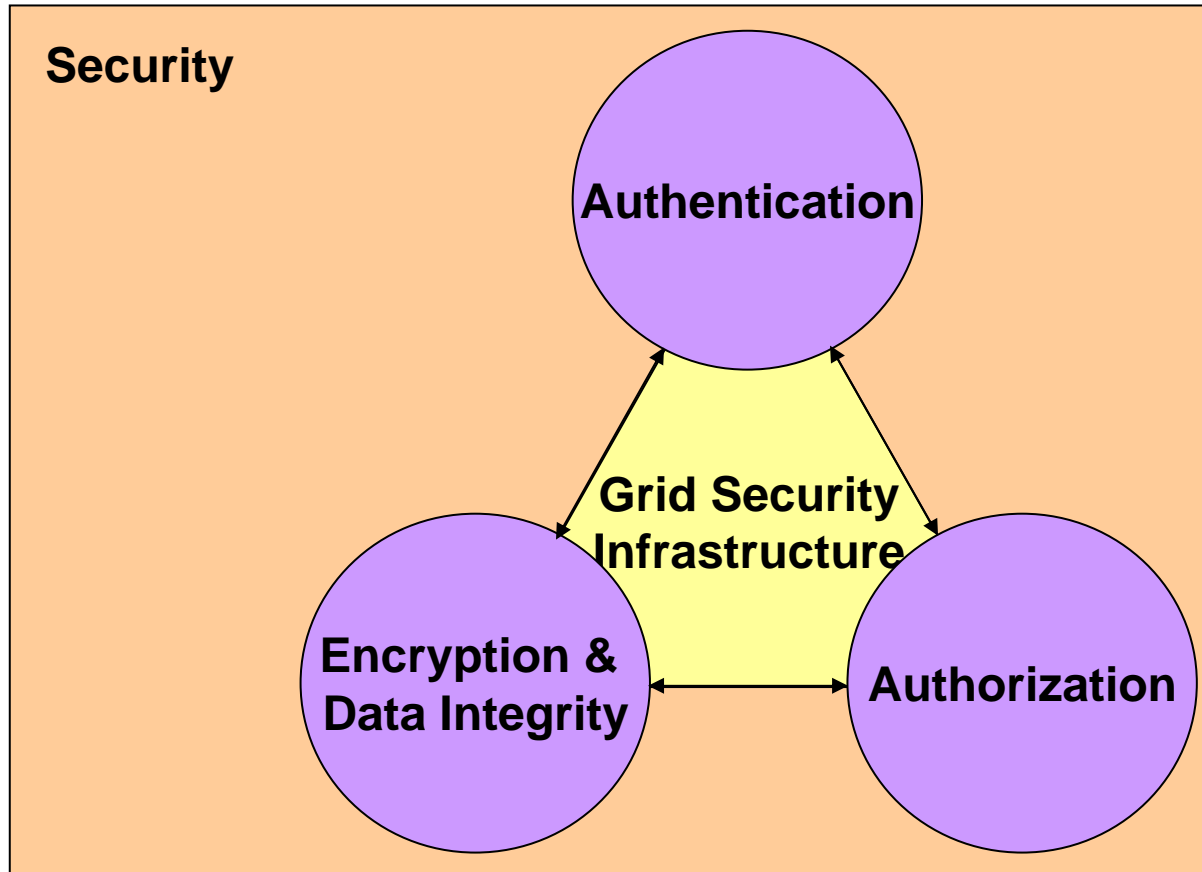mjm@nesc.ac.uk, gcw@nesc.ac.uk

Policy for re-use

- This presentation can be re-used for academic purposes.

- However if you do so then please let training-support@nesc.ac.uk know. We need to gather statistics of re-use: no. of events, number of people trained. Thank you!!
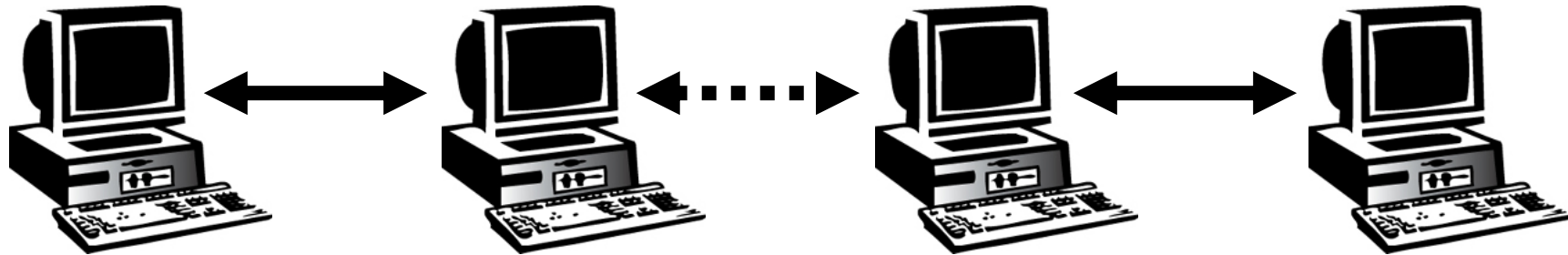
Security Overview

The Problems - 1

User   **Resource**

- How does a user securely access the Resource without having an account on the machines in between or even on the Resource?

- How does the Resource know who a user is?
- How are rights and that they are allowed access?

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**
  - Massive distributed storage capacity ideal for example, for swapping movies.

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

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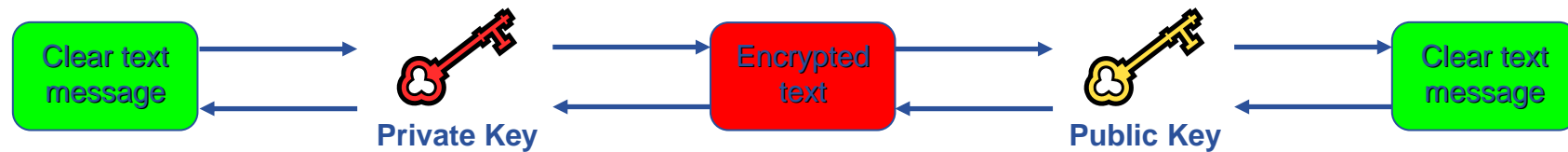
- **…. and Digital signatures …**
  - A hash derived from the message and encrypted with the signer's private key
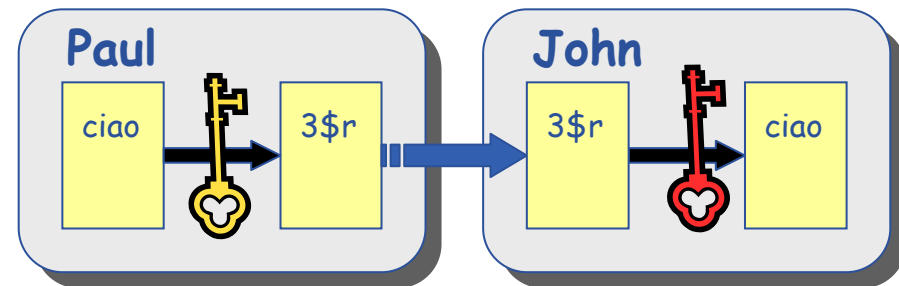 - Signature is checked by decrypting with the signer's public key

- **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

Enabling Grids for E-sciencE

- **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.
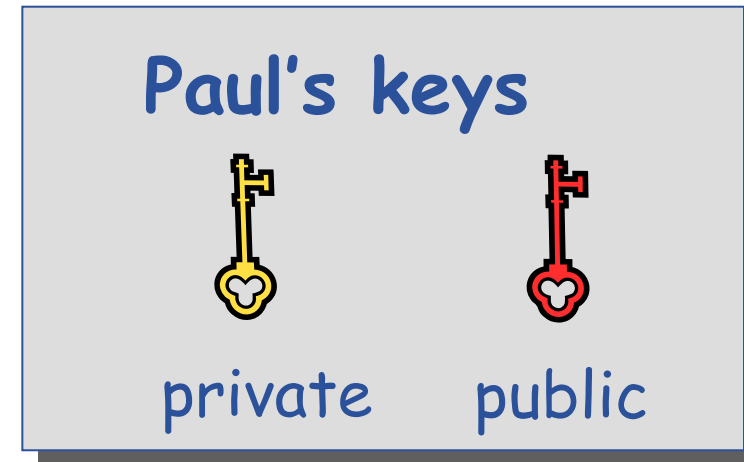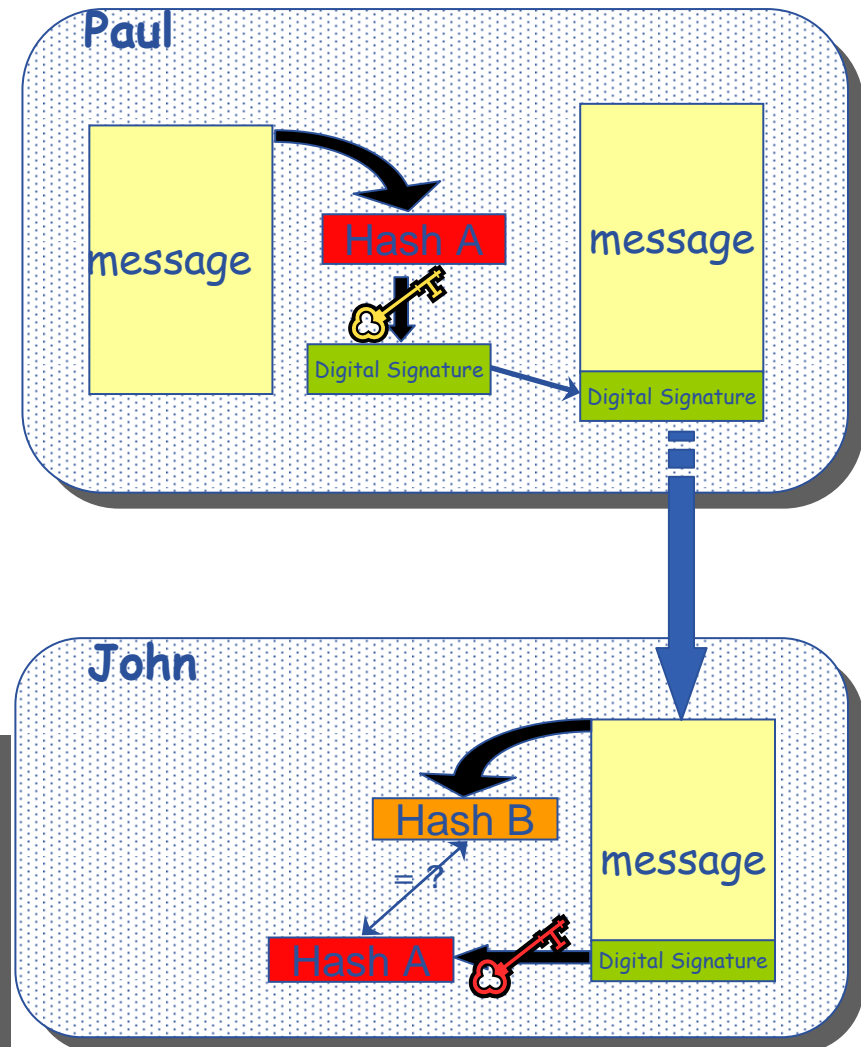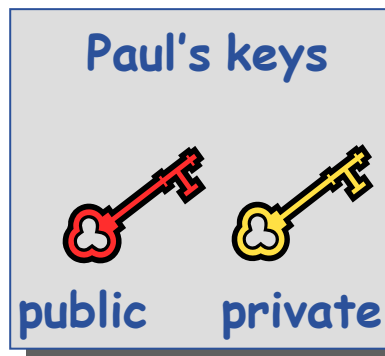
- **Public keys are exchanged**

- **The sender encrypts using his private key**

- **The receiver decrypts using senders public key;**

- **The number of keys is O(n)**

Paul's keys

private   public

Paul   John

ciao 3$r   3$r ciao

Enabling Grids for E-sciencE

- 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**

Paul
message
Hash A
Digital Signature
message
Digital Signature

Paul's keys
public   private

John
Hash B
Hash A
message
Digital Signature

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;
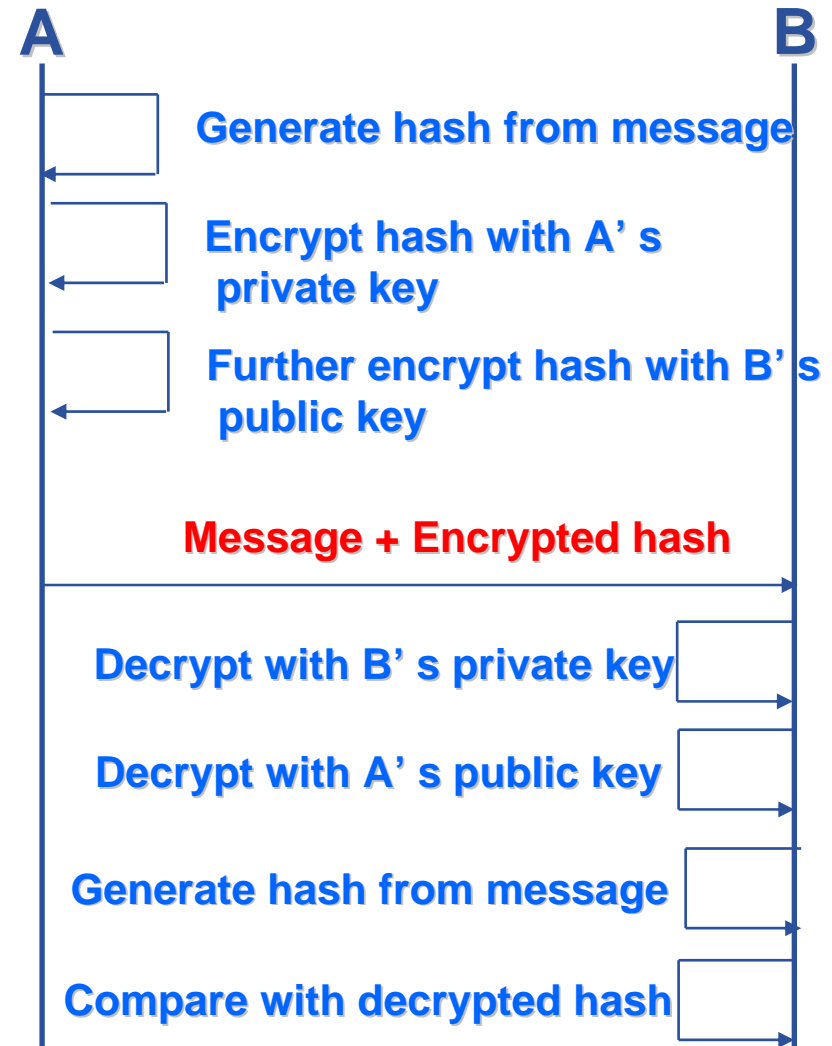   3. B sends to A a challenge string;
    4. A encrypts the challenge string with his private key;
    5. A sends encrypted challenge to B
    6. B uses A's public key to decrypt the challenge.
    7. B compares the decrypted string with the original challenge
    8. If they match, B verified A's identity and A can not repudiate it.
    9. Repeat for A to verify B's identity

A   **B**

A's certificate

Verify CA signature

Random phrase

Encrypt with A' s private key

Encrypted phrase

Decrypt with A' s public key

Compare with original phrase
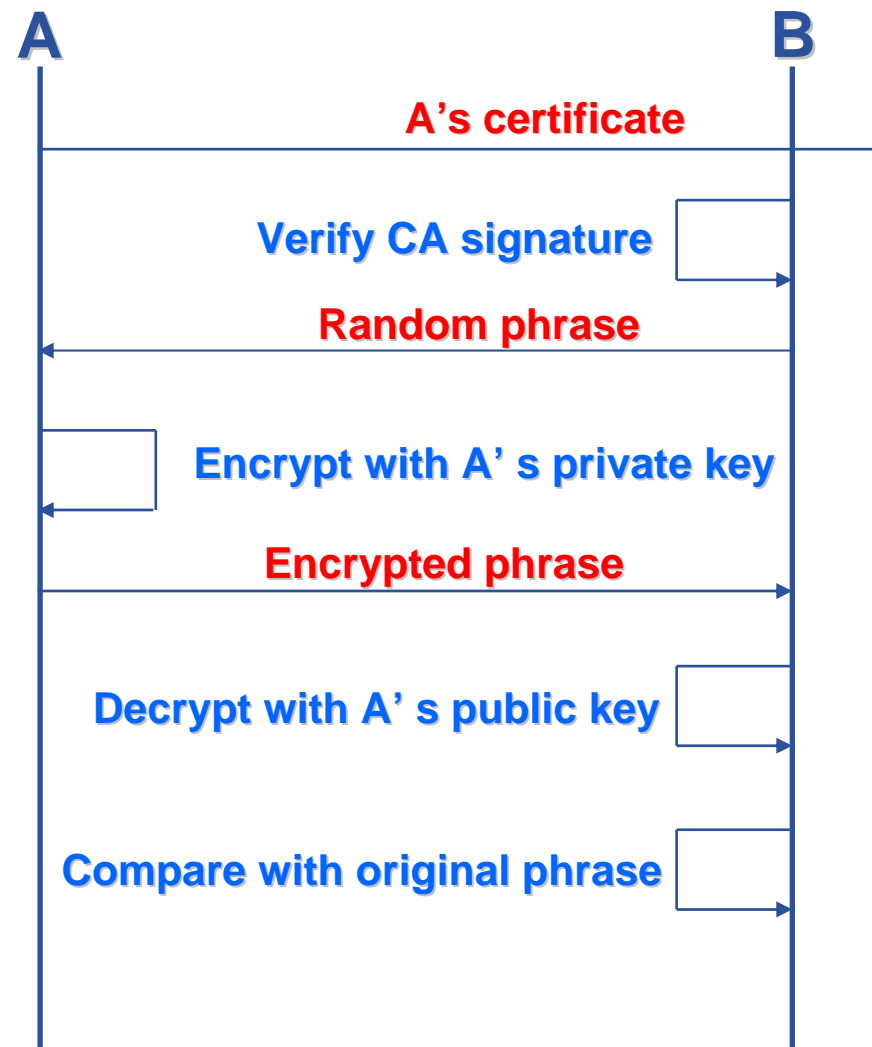
After A and B authenticated each other,
for A to send a message to B:

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

.

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

A **B**

Generate hash from message

Encrypt hash with A's private key

Further encrypt hash with B's public key

Message + Encrypted hash

Decrypt with B's private key

Decrypt with A's public key

Generate hash from message

Compare with decrypted hash
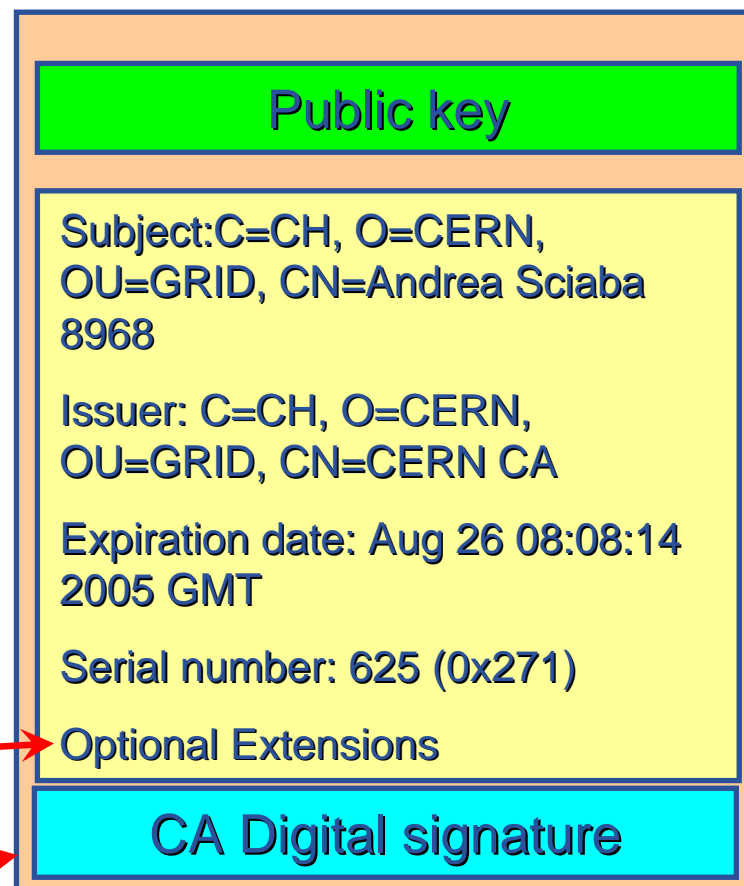
Enabling Grids for E-sciencE

- **How can John be sure that Paul's public key is really <u>Paul's</u> 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).

Enabling Grids for E-sciencE

- **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
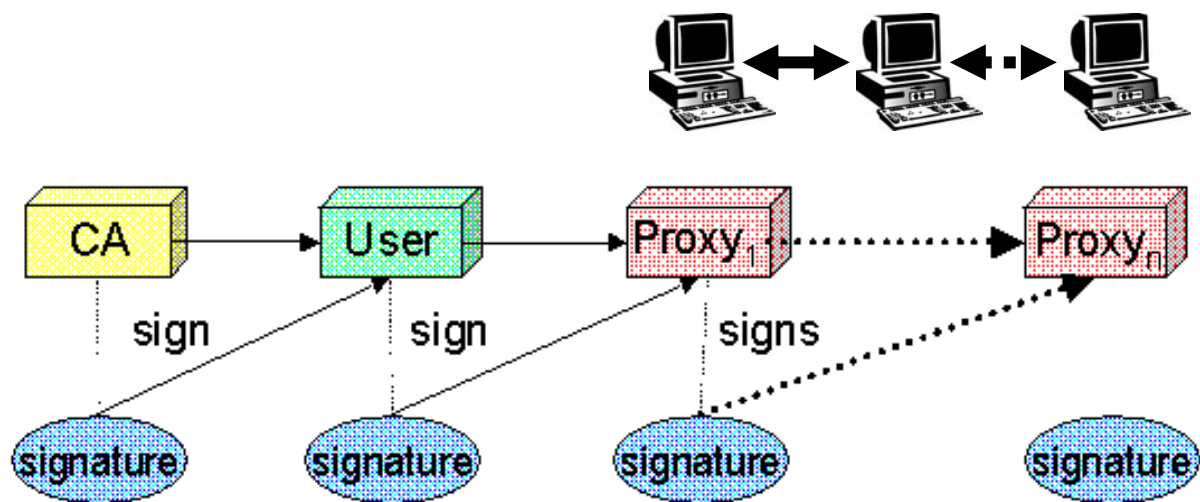

Public key

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

- **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**

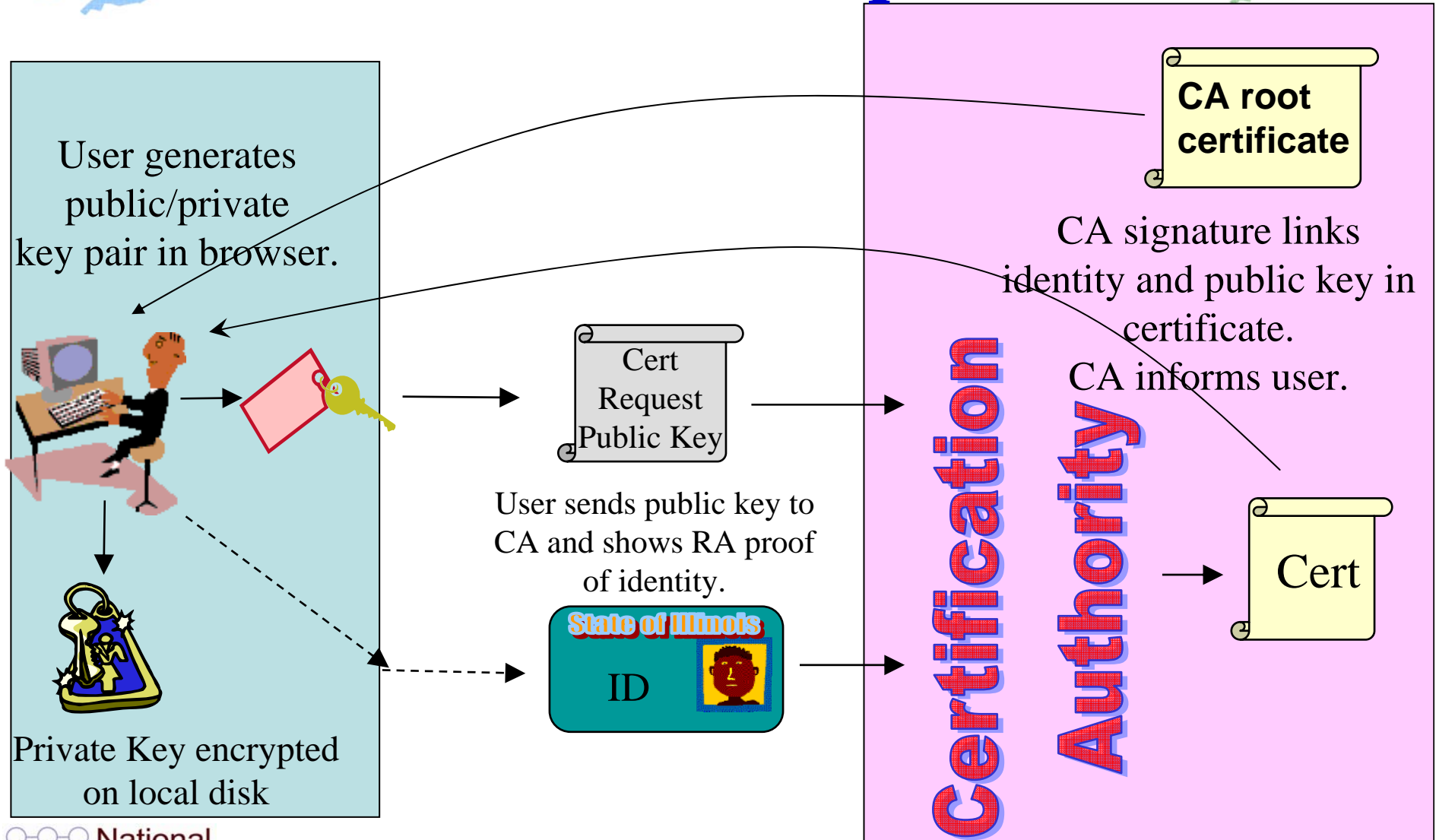- **To find RAs in UK go to http://www.grid-support.ac.uk/ca/ralist.htm**

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
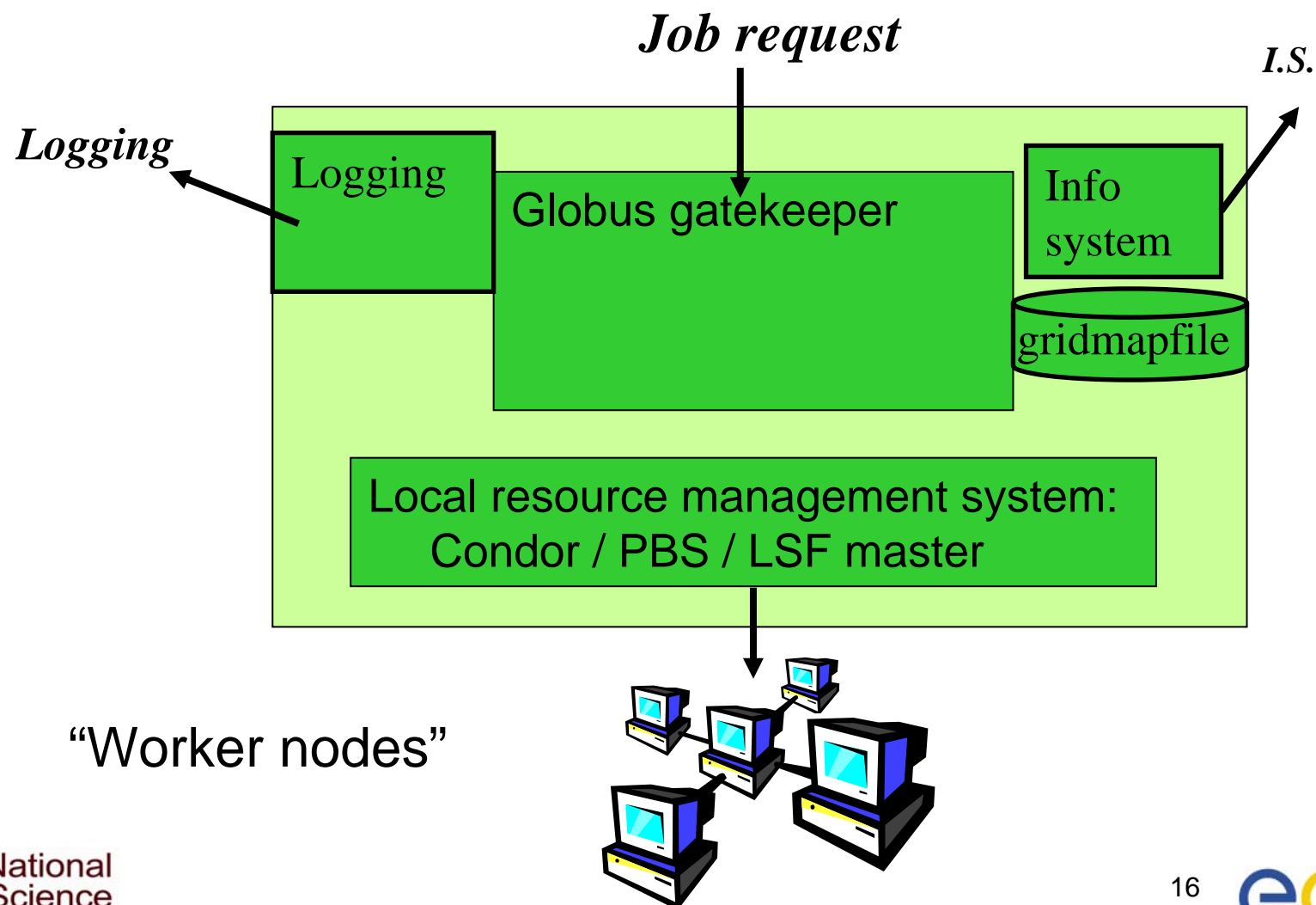 - Reduces security risk, enables delegation

Certificate Request

GOSC

NGS National Grid Service

User generates public/private key pair in browser.

Private Key encrypted on local disk

Cert Request Public Key

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

State of Illinois

ID

CA root certificate

CA signature links identity and public key in certificate.
CA informs user.

Certification Authority

Cert

National e-Science Centre

eGee

"Compute element": An LRMS queue

Job request

I.S.

Logging

Logging

Globus gatekeeper

Info system

gridmapfile

Local resource management system:
Condor / PBS / LSF master

"Worker nodes"

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.

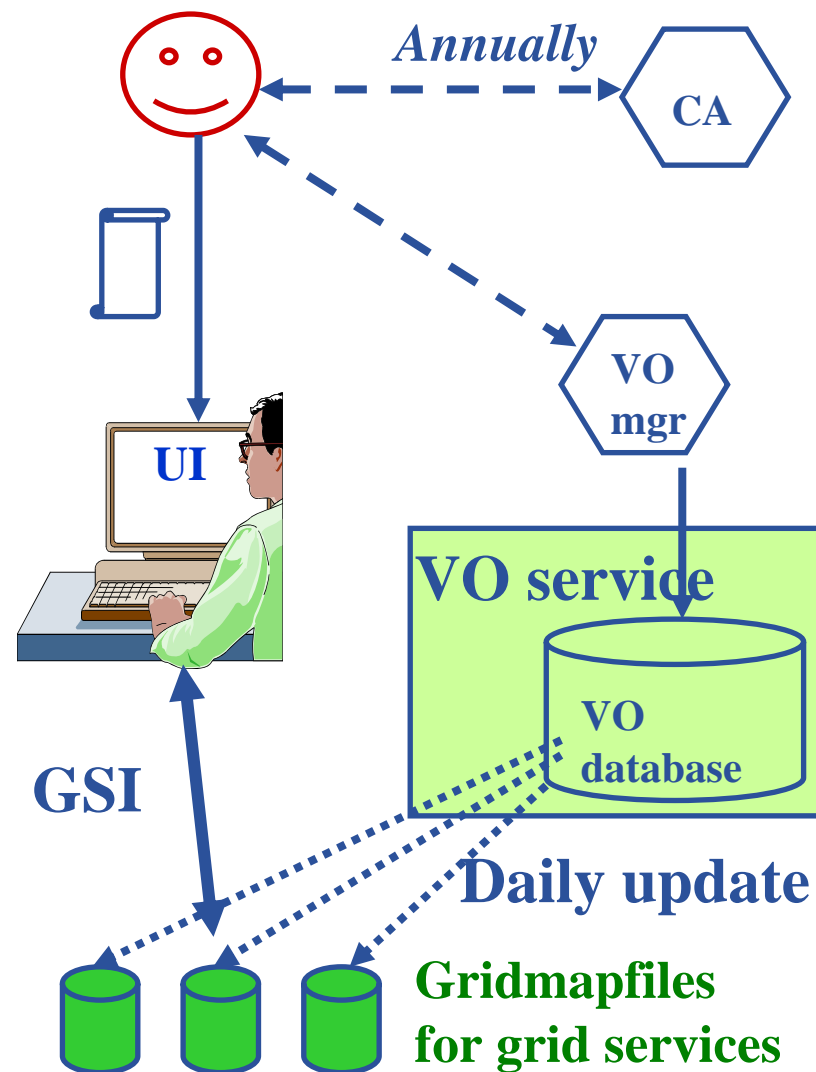- Do not launch a delegation service for longer than your current task needs.

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

- **Authentication**

  - User obtains certificate from Certificate Authority

  - Connects to UI by ssh
    UI is the user's interface to Grid

  - Uploads certificate to UI

  - Single logon – to UI - create proxy

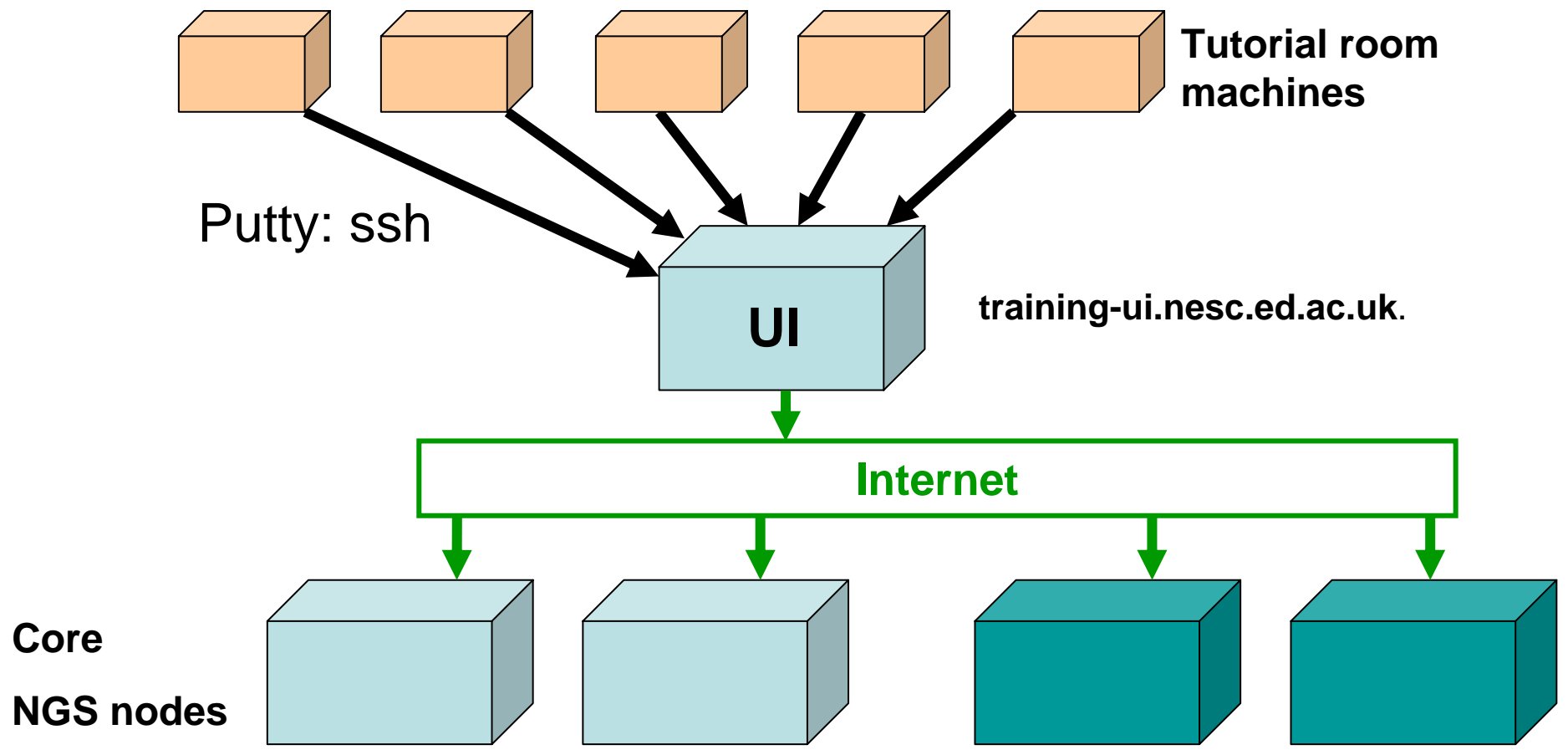 - then **Grid Security Infrastructure uses proxies**

- **Authorisation**

  - User joins Virtual Organisation

  - VO negotiates access to Grid nodes and resources

  - Authorisation tested by resource:

  Gridmapfile (or similar) maps user to local account

Annually

CA

UI

VO mgr

VO service

VO database

GSI

Daily update

Gridmapfiles for grid services

Our setup

Tutorial room machines

Putty: ssh

UI

training-ui.nesc.ed.ac.uk.

Internet

Core

NGS nodes

The Practical

- You should have been given an information sheet containing your username and password

1. Login to your workstation
2. Open a browser window and follow the link from http://agenda.cern.ch/fullAgenda.php?ida=a061956
3. Click on the "further information" for *this* practical. …. Find Putty on your PC
4. Open a new session on putty:
   1. connect to training-ui.nesc.ed.ac.uk
   2. Accept key when prompted by training-ui

The Practical

- You should have been given an information sheet containing your username and password

1. Login to your workstation
2. Open a browser window and follow the link from http://agenda.cern.ch/fullAgenda.php?ida=a061881
3. Click on the "further information" for *this* practical.
4. Before you follow instructions to use the putty program set it up to use X11 (useful later on).... Next slide

How to start putty to enable x11

1. Run exceed

2. Run putty

3. Set X11 before opening session

4. (kwrite editor available)

- Open a new session on putty:
- connect to training-ui.nesc.ed.ac.uk
  - Accept key when prompted by training-ui