GRID & e-Science:

Security in the Grid

Overview

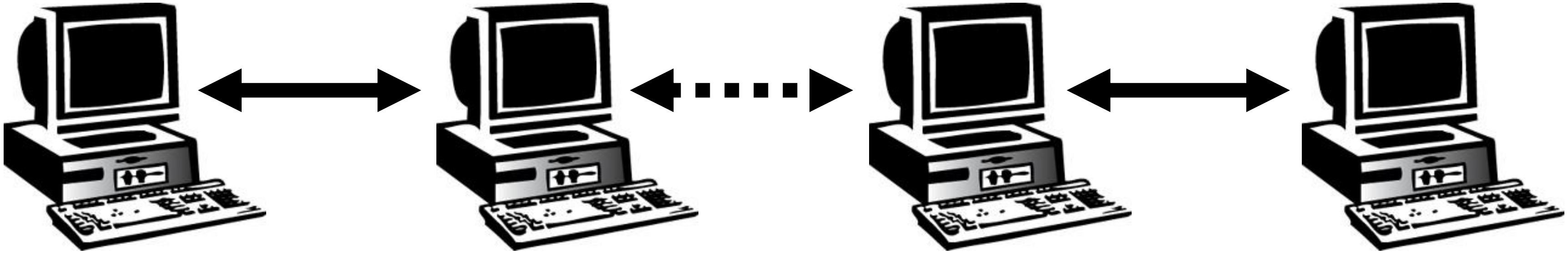
- Problems
- Glossary
- Encryption
  - Symmetric algorithms
  - Asymmetric algorithms: Public Key Infrastructure
- Certificates
  - Digital Signatures
  - X.509 certificates
- Grid Security
  - Proxy certificates
  - Command line interfaces
- Virtual Organization
  - Concept of VO and authorization

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 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

- The “Grid Security Infrastructure(GSI)” is the basis of (most) production grids

Problems



User

Resource

- How does a user securely access the Resource without having an account on the machines of the Resource?

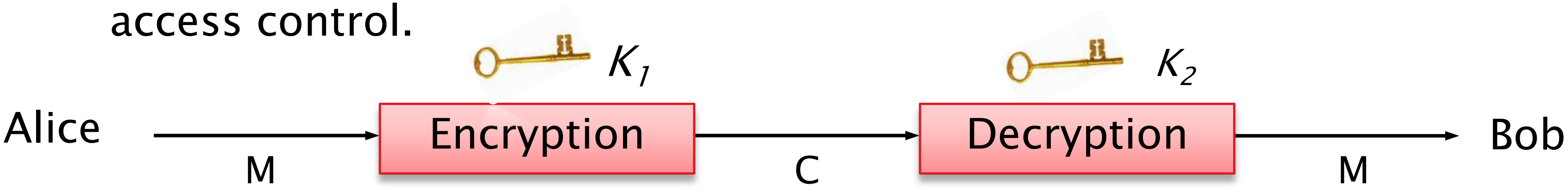
- How does the Resource know who a user is? Authentication
- How are rights and that they are allowed access? Authorization

Problems

- Security!!!
  - 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.

Cryptography

- Is a discipline of mathematics concerned with information security and related issues, particularly encryption, authentication, and access control.



K_1 K_2

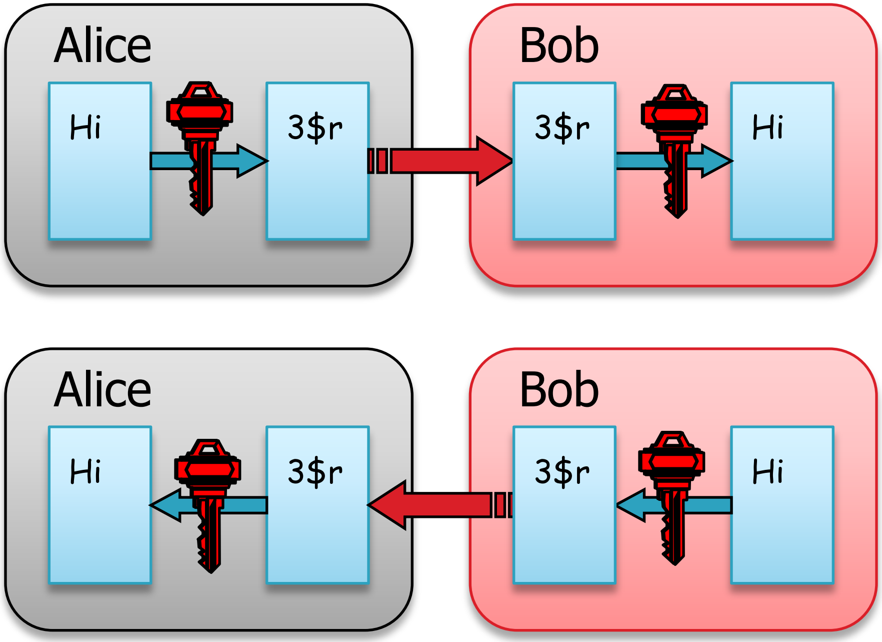
Alice Encryption Decryption Bob

M C M

- Symbology
  - **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**: $K_1 \neq K_2$

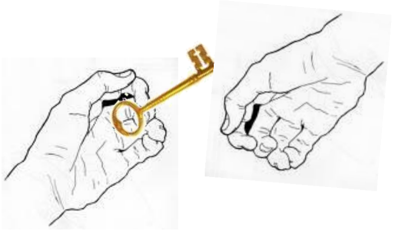
Symmetric Algorithm

- The same key is used for encryption and decryption
- The key is shared by both side of the communication



Symmetric Algorithm

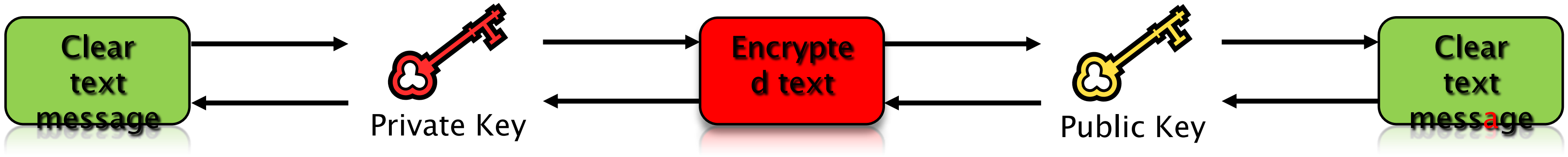
- Advantages:
  - Fast & Easy
- Problems:
  - How to distribute the key?



  - The number of keys needed is $O(n^2)$
- Examples:
  - DES (Digital Encryption Standard)
  - 3DES (Triple DES)
  - AES (Digital Encryption Standard)
  - Blowfish

Asymmetric Algorithms

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



Clear text message

Private Key

Encrypte d text

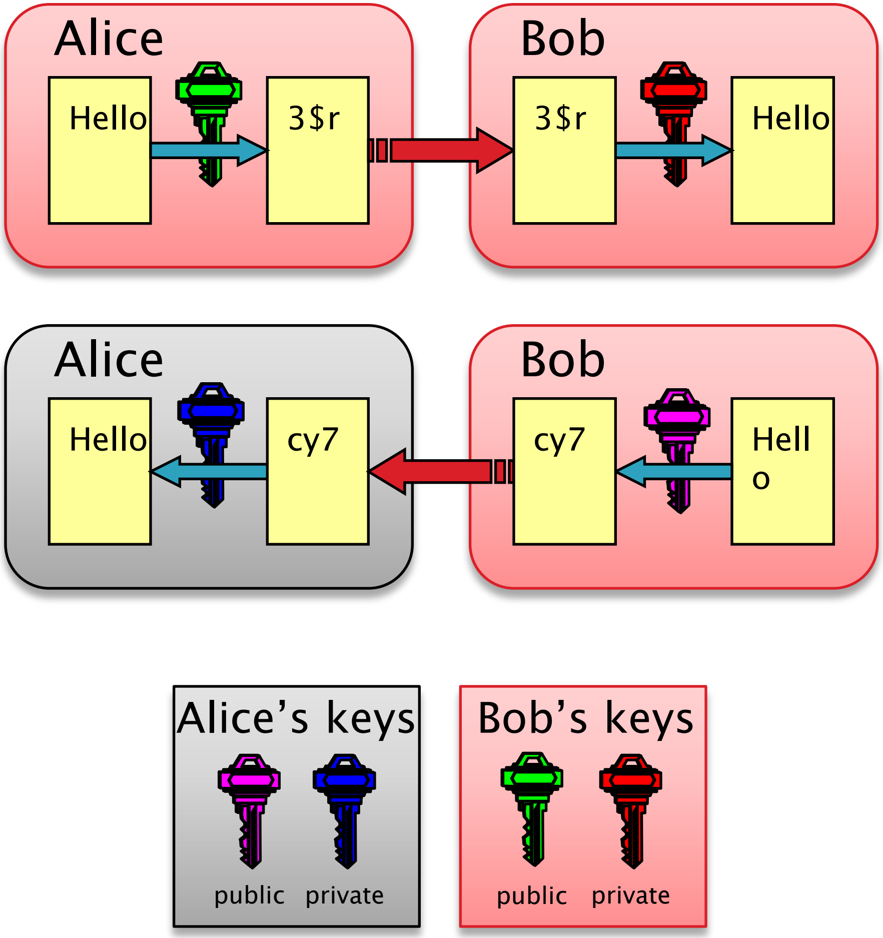
Public Key

Clear text message

  - 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 ciphers using the *public* key of the receiver
- The receiver decrypts using his *private* key;
- The number of keys is *O(n)*
- Examples:
  - Diffie-Helmann
  - RSA

Asymmetric Algorithm



Hash Function

- Converts any size of input into a fixed (smaller) size of output
  - Given *h(x)*, it is difficult to compute *x*.
  - Given *x*, it is difficult to find x' such that *h(x) = h(x').*
- Usage
  - Verifying file integrity
  - Digitally Signature
- Examples
  - MD5
  - SHA-1

Digital Signature

- 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

- Alice calculates the hash A of the message
- Alice encrypts the hash using his private key: the encrypted hash is the digital signature.
- Alice sends the signed message to Bob.
- Bob calculates the hash B of the message
- Decrypts signature, to get hash A , using Alice’s public key.
- If hashes equal:
  - message wasn’t modified;
  - hash A is from Alice’s private key

Certification Authorities

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

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

Certification Authorities

- User’s identity has to be certified by one of the national *Certification Authorities* (CAs)

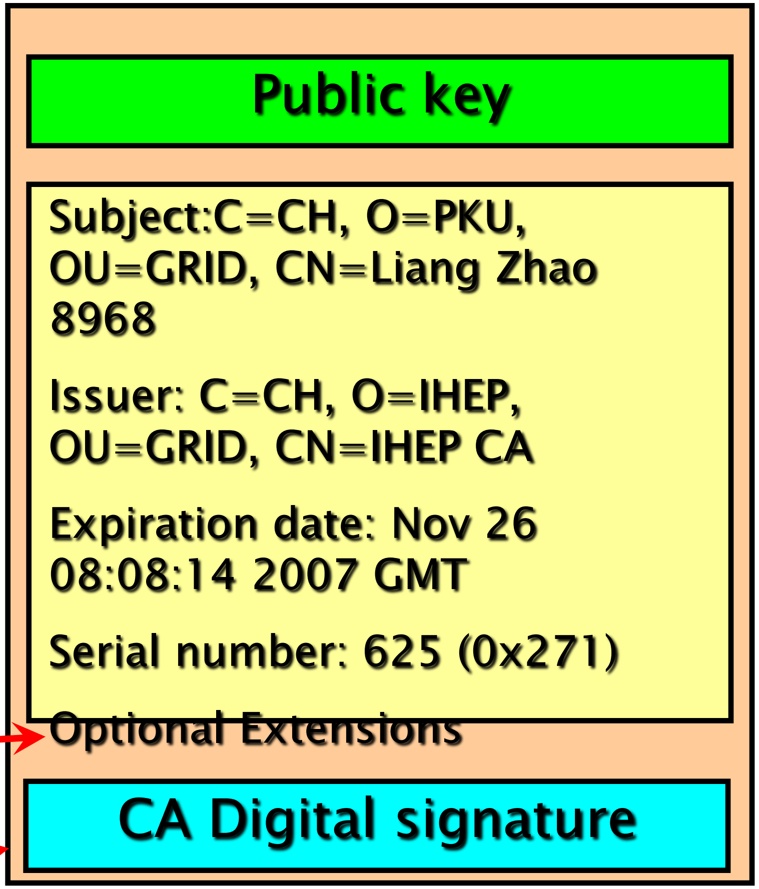
- Resources are also certified by CAs

- CAs are mutually recognized http://www.gridpma.org/

- CAs each establish a number of people “registration authorities” RAs

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=PKU, OU=GRID, CN=Liang Zhao 8968

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

Expiration date: Nov 26 08:08:14 2007 GMT

Serial number: 625 (0x271)

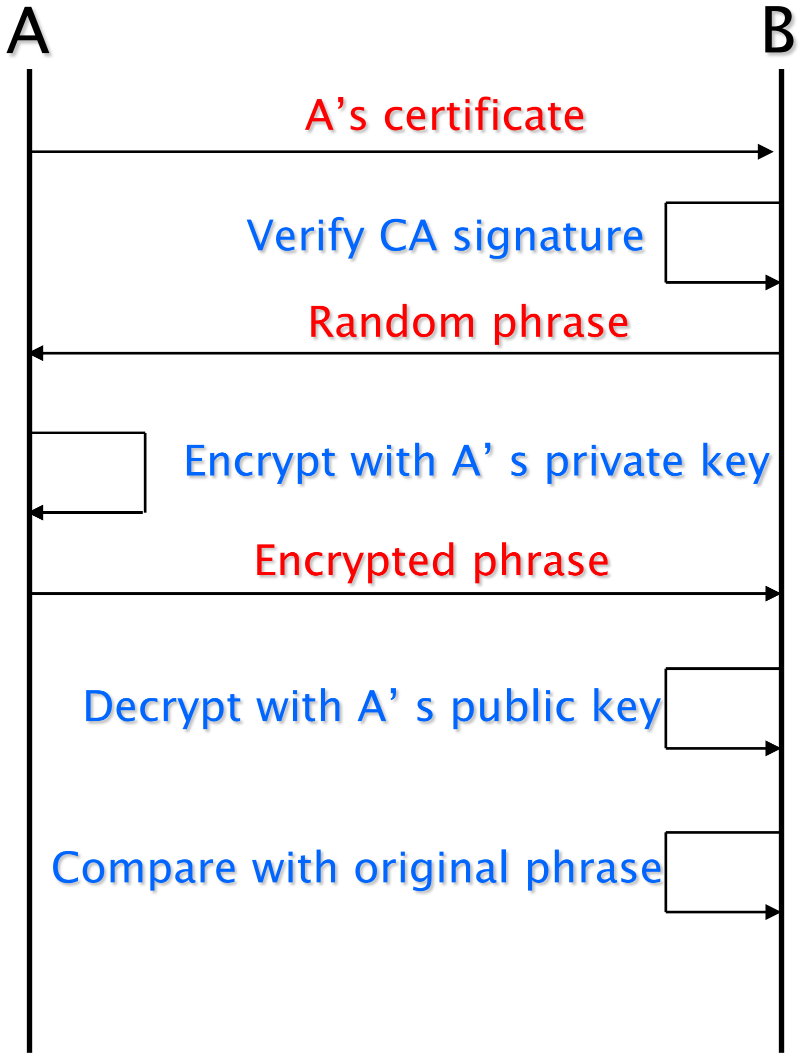
Optional Extensions

CA Digital signature

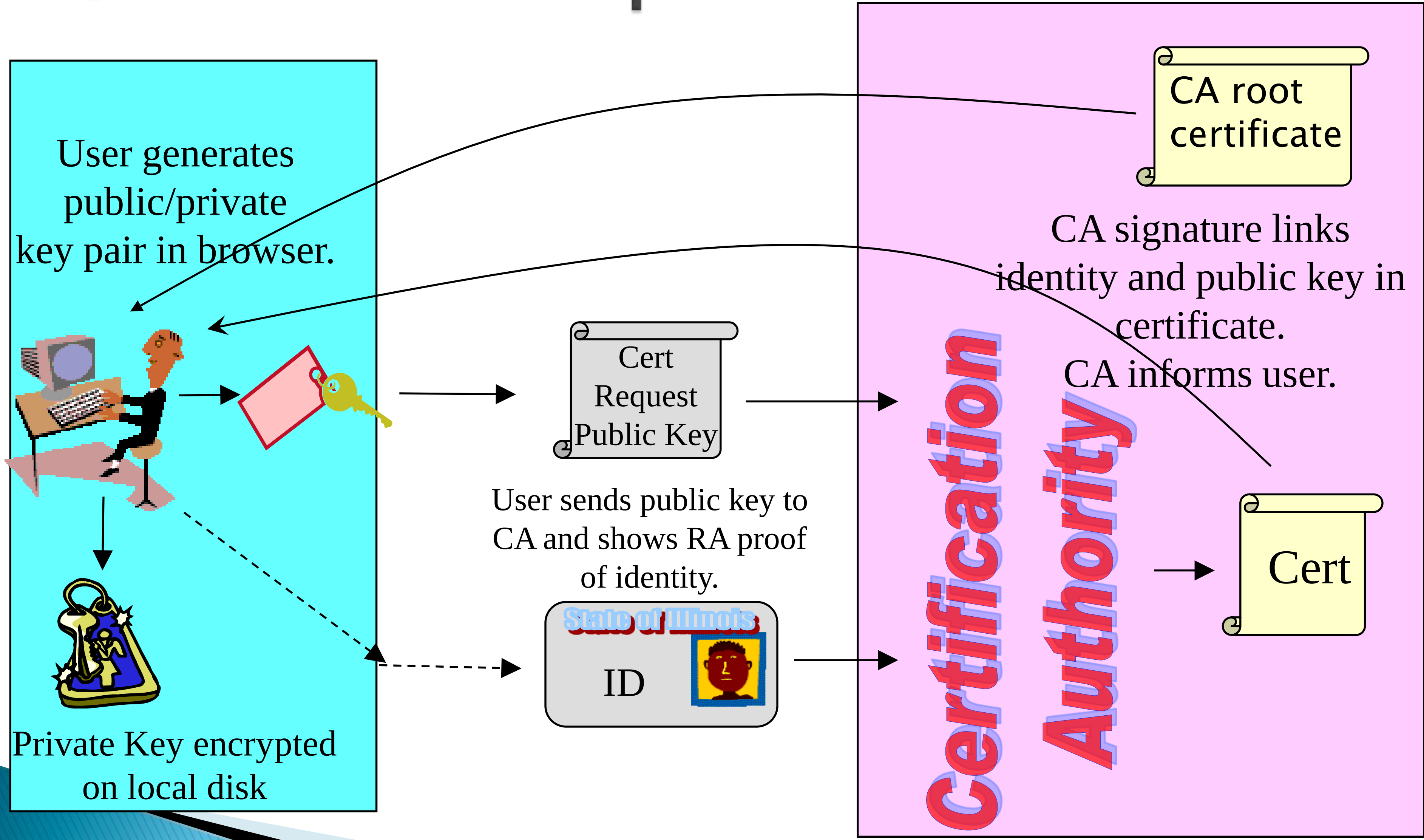
The Grid Security Infrastructure

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

Certificate Request



User generates public/private key pair in browser.

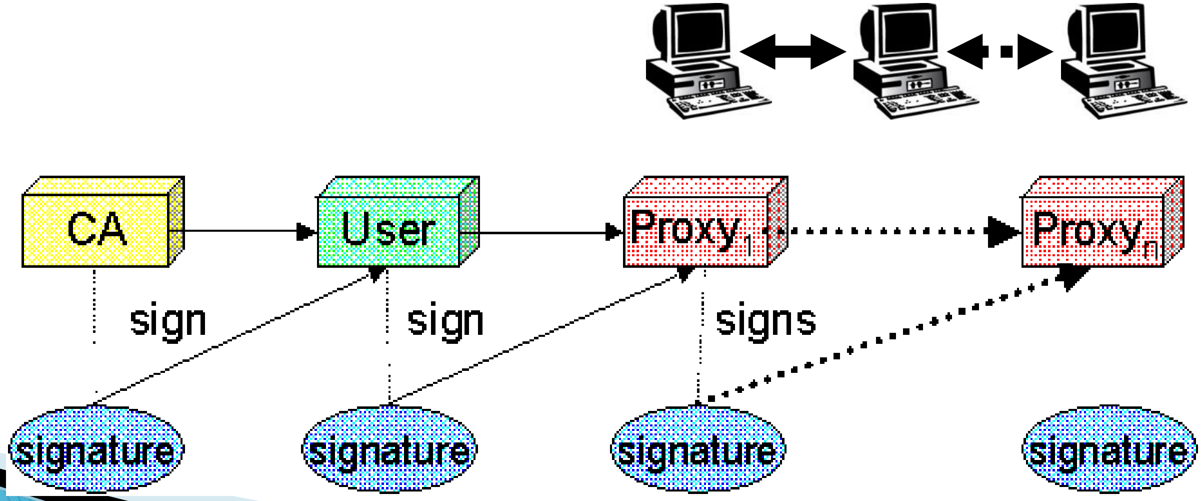
Private Key encrypted on local disk

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

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

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



User Responsibilities

- Keep your private key secure – *on USB drive only*
- 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.

Evolution of VO management

Before VOMS

- User is authorized 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`

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`

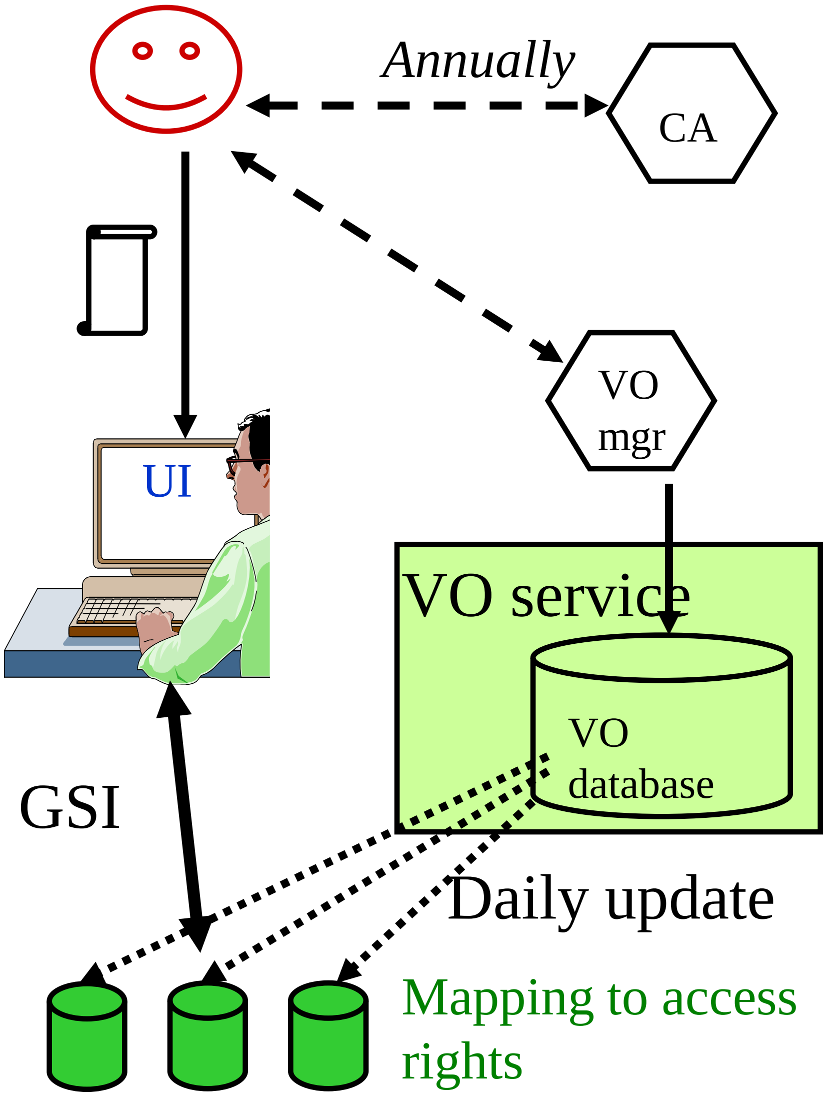
VOMS – now in use on EGEE grid

Summary 1

- Authentication based on X.509 PKI infrastructure
  - Trust between Certificate Authorities (CA) and sites, CAs and users is established (offline)
  - CAs issue (long lived) certificates identifying sites and individuals (much like a passport)
    - Commonly used in web browsers to authenticate to sites
  - In order to reduce vulnerability, on the Grid user identification is done by using (short lived) proxies of their certificates
- Proxies can
  - Be delegated to a service such that it can act on the user's behalf
  - Include additional attributes (like VO information via the VO Membership Service VOMS)
  - Be stored in an external proxy store (MyProxy)
  - Be renewed (in case they are about to expire)

Summary 2

- 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
  - **Grid Security Infrastructure**
- Authorisation
  - User joins Virtual Organisation
  - VO negotiates access to Grid nodes and resources
  - Authorisation tested by resource:
    - Credentials in proxy determine user's rights



Mapping to access rights