Grid Security (1)

Grid Computing (M)

Richard Sinnott

UNIVERSITY
of
GLASGOW

Why is Grid security so important?

- If it is not secure
  - Large communities will not engage
    - medical community, industry, financial community …
  - Legal and ethical issues possible to be violated with all sorts of consequences
    - e.g. data protection act violations and fines incurred
  - Expensive (impossible?) to repeat some experiments
    - Huge machines running large simulations for several years
  - Trust (more later) is easily lost and hard to re-establish
  - Grid resources are a dream for hackers
    - Huge file storage for keeping their "dodgy data"
    - Perfect environment for launching attacks like distributed denial of service
      - Not just access to one machine
        - » whole interconnected networks of ultra performant machines which can be used for cracking passwords, codes, launching distributed denial of service attacks, …

The Challenge of Grid Security

- Grids allow (or should allow!) dynamic establishment of virtual organisations (VOs)
  - These can be arbitrarily complex
    - Grids (VOs) might include highly secure supercomputing facilities through to single user PCs/laptops
    - Need security technologies that scales to meet wide variety of applications
      - from highly secure medical information data sets through to particle physics/public genome data sets
      - Using services for processing of patient data through to “needle in haystack” searching of physics experiments or protein sequence similarity of genomic data
  - Should try to develop generic Grid security solutions
    - Avoid all application areas re-inventing their own (incompatible/inoperable) solutions

The Challenge of Grid Security …ctd

- Grid allows scenarios that stretch inter-organisational security
  - Imagine two distributed virtual organisations agreeing to share resources, e.g. compute/data resources to accomplish some task with sharing done across internet
    - Could have policies that restrict access to and usage of resources based on pre-identified users, resources
    - But what if new resources added, new users added, old users go,…?
    - What if organisations decide to change policies governing access to and usage of resources?
    - What if want to transfer large data sets between different organisations – how to ensure that data is not cached somewhere it might be compromised?
    - …

Prelude to Grid Security

- What do we mean by security anyway?
  - Secure from whom?
    - From sys-admin?
    - From rogue employee?
    - …
  - Secure against what?
    - Security is never black and white but is a grey landscape where the context determines the accuracy of how secure a system is
      - e.g. secure as given by a set of security requirements
  - Secure for how long?
    - *"I recommend overwriting a deleted file seven times: the first time with all ones, the second time with all zeros, and five times with a cryptographically secure pseudo-random sequence. Recent developments at the National Institute of Standards and Technology with electron-tunnelling microscopes suggest even that might not be enough. Honestly, if your data is sufficiently valuable, assume that it is impossible to erase data completely off magnetic media. Burn or shred the media; it's cheaper to buy media new than to lose your secrets…."*

      » *-Applied Cryptography 1996, page 229*

Prelude to Grid Security …ctd

- Note that security technology ≠ secure system
  - Ultra secure system using 2048+ bit encryption technology, packet filtering firewalls, …
    - …. on laptop in unlocked room
    - … on PC with password on "post-it" on screen/desk
    - …
  - Famous quote to muse over:
    - *"…if you think that technology can solve your security problems then you don't know enough about the technology, and worse you don't know what your problems are…"*
      - » *Bruce Schneier, Secrets and Lies in a Digital Networked World*

Technical Challenges of Grid Security

- Key terms that are typically associated with security
  - Authentication
  - Authorisation
  - Audit/accounting
  - Confidentiality
  - Privacy
  - Integrity
  - Fabric management
  - Trust

All are important for Grids but some applications may have more emphasis on certain concepts than others

Security Concepts::Authentication

- Authentication is the establishment and propagation of a user's identity in the system
  - e.g. so site X can check that user Y is attempting to gain access to resources
    - Note does not check what user is allowed to do, only that we know (and can check!) who they are
      - Masquerading always a danger (and realistic possibility)
      - Need for user guidance on security
        - Password selection
        - Treatment of certificates (more later)
        - …

- Typically achieved using Public Key Infrastructures
  - More later…

Security Concepts::Authorisation

- *Authorisation*
  - concerned with controlling access to services based on policy
    - Can this user invoke this service making use of this data?
    - Complementary to authentication
      - Know it is this user, now can we restrict/enforce what they can/cannot do
  - Many different contenders for authorisation infrastructures
    - PERMIS
    - CAS
    - VOMS
    - AKENTI
    - VOM
  - …we have worked extensively with PERMIS explored later and in assignment

Security Concepts::Auditing

- *Auditing*
  - the analysis of records of account (e.g. security event logs) to investigate security events, procedures or the records themselves
    - Includes logging, intrusion detection and auditing of security in managed computer facilities
      - well established in theory and practice
        - » Grid computing adds the complication that some of the information required by a local audit system may be distributed elsewhere, or may be obscured by layers of indirection
        - » e.g. Grid service making use of federated data resource where data kept and managed remotely
    - Need tools to support the generation of diagnostic trails
      - Do we need to log all information?
      - How long do we keep it for?
      - …

Security Concepts::Confidentiality

- *Confidentiality*
  - is concerned with ensuring that information is not made available to unauthorised individuals, services or processes
    - It is usually supported by access control within systems, and encryption between systems
      - Confidentiality is generally well understood, but the Grid introduces the new problem of transferring or signalling the intended protection policy when data staged between systems

Security Concepts::Privacy

- *Privacy*
  - particularly significant for projects processing personal information, or subject to ethical restrictions
    - e.g. projects dealing with medical, health data
  - Privacy requirements relate to the use of data, in the context of consent established by the data owner
    - Privacy is therefore distinct from confidentiality, although it may be supported by confidentiality mechanisms.
    - Grid technology needs a transferable understanding of suitable policies addressing privacy requirements/constraints
      - Should allow to express how such policies can be
        - » defined,
        - » applied,
        - » implemented,
        - » enforced, …

Security Concepts::Integrity

- *Integrity*
  - **Ensuring that data is not modified since it was created, typically of relevance when data is sent over public network**
    - Technical solutions exist to maintain the integrity of data in transit
      - Explore some of these in PKIs
    - Grid also raises more general questions
      - e.g. provenance
        - » maintaining the integrity of chains or groups of related data

Security Concepts::Fabric Management

- *Fabric Management*
  - **consists of the distributed computing, network resources and associated connections that support Grid applications**
    - impacts Grid security in two ways:
      - an insecure fabric may undermine the security of the Grid
      - fabric security measures may impede grid operations
        - » e.g. firewalls may be configured to block essential Grid traffic

Security Concepts::Trust

- *Trust*
  - **characteristic allowing one entity to assume that a second entity will behave exactly as the first entity expects**
  - **Important distinction between 'trust management' systems which implement authorisation, and the wider requirements of trust**
    - e.g. health applications require the agreement between users and resources providers of restrictions that cannot be implemented by access control
      - e.g. restrictions on the export of software, or a guarantee that personal data is deleted after use
    - therefore a need to understand and represent policy agreements between groups of users and resource providers
      - such policies may exist inside or outside the system, and are typically not supported by technical mechanisms

Grid Security Basics

- We want all of the above, but…
  - Little consensus on most concepts
  - Best practice to copy

- Main area of agreement and adoption by Grid community is idea of Public Key Infrastructure (PKI)

Introduction to PKI

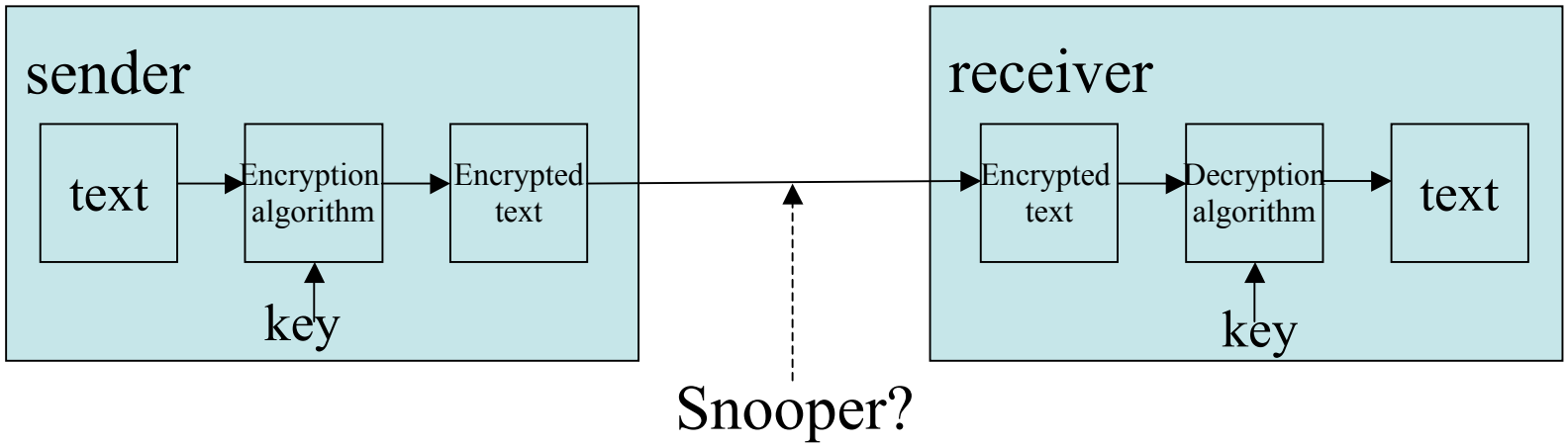
- In the beginning of the internet security was not prime concern
  - No longer the case
    - Ever growing dependencies on security over internet
      - Banking, finances, shopping, …
  - Question is how do we implement it?
    - Collection of approaches, standards, solutions, …
  - Public Key Infrastructures (PKI) offer one possibility
  - Offer many advantages to Grid community
    - (more later)

Brief Recap on Cryptography

- Word *cryptography* means hidden or secret writing
  - Scrambling (encrypting)/unscrambling (decrypting) private messages to support *confidentiality*
- Cryptographic algorithms allow to check if message has been changed since it was created/sent and to identify message sender
  - Message that is unaltered said to have *integrity*
- Algorithm defines the steps for sender to scramble message and receiver to unscramble message

Cryptography Algorithms

- Various types of algorithms exist
- Most use two inputs
  - A message to be encrypted/decrypted
  - A *key*
- Symmetric Cryptography
  - Sender and receiver use same key value
    - Also called shared secret key systems
  - Best keys are long random bit strings



Symmetric Cryptography

- *Snooper* wants to access the ***text***
- To do this they need the key
  - Could try brute force all combinations
    - Not realistic for large (1024+ bit)
      - ...or (possibly????) wasn't until arrival of large compute resources
        - » e.g. those available on Grids
- If sender/receiver want to share with numerous other senders/receivers then likely to need many keys
- Key issue is thus *key management*
  - Need to support creation, distribution, use, archival, destruction of keys
    - Must ensure storage/archival protects from unwanted disclosure
    - Keys remain associated with correct senders/receivers

Asymmetric Cryptography

- Also called Public Key Cryptography
  - Has two distinct keys
    - One that must be kept private
      - Private Key …. Duh! ;o)
    - One that can be made public
      - Public Key … Double duh!
  - Two keys are complementary, but essential that cannot find out value of private key from public key
    - With private keys can digitally sign messages, documents, … and validate them with associated public keys
      - Check whether changed, useful for non-repudiation, …
- Public Key Cryptography greatly simplifies management of keys:
  - Don't need to have many keys for long time
    - The longer keys are left in storage, more likelihood of their being compromised
      - Instead use Public Keys for short time and then discard
      - Public keys can be freely distributed
    - Only Private Key needs to be kept long term and kept securely

Public Key Algorithms

- Two key management public key algorithm types
  - Key agreement
    - Public keys exchanged and private keys used with the corresponding public key to create symmetric key known only to both parties (parties have prior agreement on the key generator)
      - Most well known algorithm for this is Diffie-Hellman, 1976
  - Key transport
    - Sender (or receiver) creates symmetric key and encrypts it with respectively the receiver (or sender) public key
    - Receiver (or sender) then decrypts using their own private key
      - Most well known algorithm for this is RSA (Rivest-Shamir-Adleman, 1978)

Public Key Certificates

- Both Public Key Agreement/Transport need to know who remote public key belongs to
  - i.e. who has associated private key
    - Otherwise share symmetric key with unknown party
- Need mechanism connecting public key to user with corresponding private key
  - This is role of *Public Key Certificate*
    - Public key certificate contains public key and identifies the user with the corresponding private key
  - Not a new idea
    - Business card
      - My name, my association, contact details, …
        - » Can be distributed to people I want to exchange info with
      - If include public key on it, then have basic certificate, but …
        - » has to be delivered in person (or no trust!), who says I work at Glasgow?, could be a forgery, I might be an impostor, what if I move to Edinburgh or my phone number changes, who would have 1024-bit key on business card, …

Public Key Infrastructures (PKI)

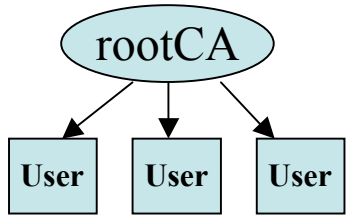
- Public Key Infrastructure (PKI) responsible for deciding policy/managing, enforcing certificate validity checks
- Central component of PKI is *Certificate Authority* (CA)
  - CA has numerous responsibilities
    - Issuing certificates
      - Often need to delegate to local Registration Authority
        - » Prove who you are, e.g. with passport
    - Revoking certificates
      - Certificate Revocation List (CRL) for expired/compromised certificates
    - Storing, archiving
      - Keeping track of existing certificates, various other information, …
  - CA often (but not always) is trusted organisation to you/your organisation
    - UK e-Science has CA in Rutherford Appleton Labs
      - Strict, policies and procedures for getting Grid certificates

PKI Trust Model

- CA issues certificates
  - Could be to users, resources, other CAs, …
    - CA certificates can describe/limit trust relationship
- Issuing certificate is indication of trust
  - CA trusts it is really you who is applying for and going to use this certificate
  - You (and others using this CA) trust that certificates are managed correctly
- How to decide if CA is trustworthy?
  - Different choices
    - User decides to trust CA
    - CAs decide if they trust one another
      - *Certification paths* used to track trust relationships
- Different architectural choices for PKI impact upon certification paths and validity checking

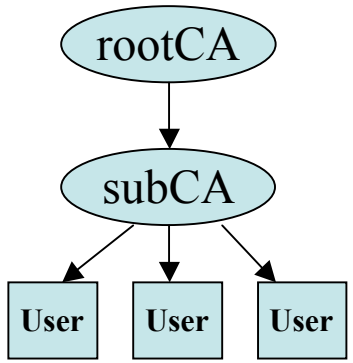
PKI Architectural Choices

- Various PKI architectures possible
  - Depends largely on trust relationships
    - How many CAs trusted?
    - How important to be able to add new CAs?
    - What kind of trust relationships between CAs?
    - …
  - Single CA
    - All users trust their own single CA
      - Users only accept certificates, CRLs issued by their CA
    - Certificate path analysis easy here
      - All certificates are user certificates
    - But, …
      - Doesn't scale to large diverse communities
      - CA is single point of failure
      - Compromised CA =
        - » ALL users contacted
        - » ALL certificates revoked



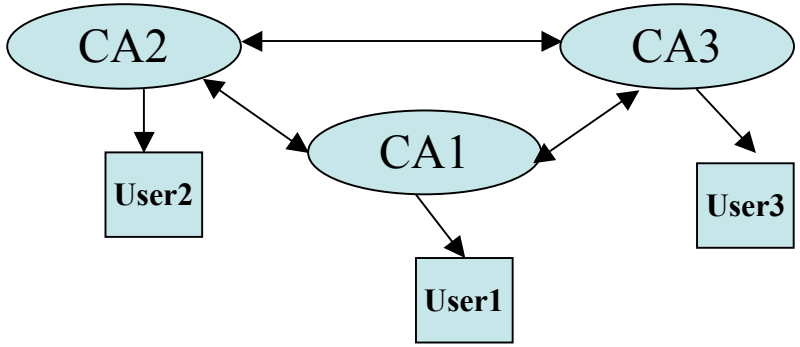
PKI Architectural Choices …ctd

- Other more complex architectures exist
  - Trust Lists
    - Users keep list of trusted CAs
      - How to tell trustworthy one from untrustworthy one?
      - Also problem of compromised CAs
        - » User not necessarily informed if directly trusted CA compromised
  - Hierarchical PKIs
    - Chains of trust with single root
      - Allow to limit damage of compromised subordinate CAs
      - Can limit actions of subordinate CAs
      - But, …
        - » Compromised root = all certificates reissued
        - » …possibly by many subordinate CAs



PKI Architectural Choices …ctd

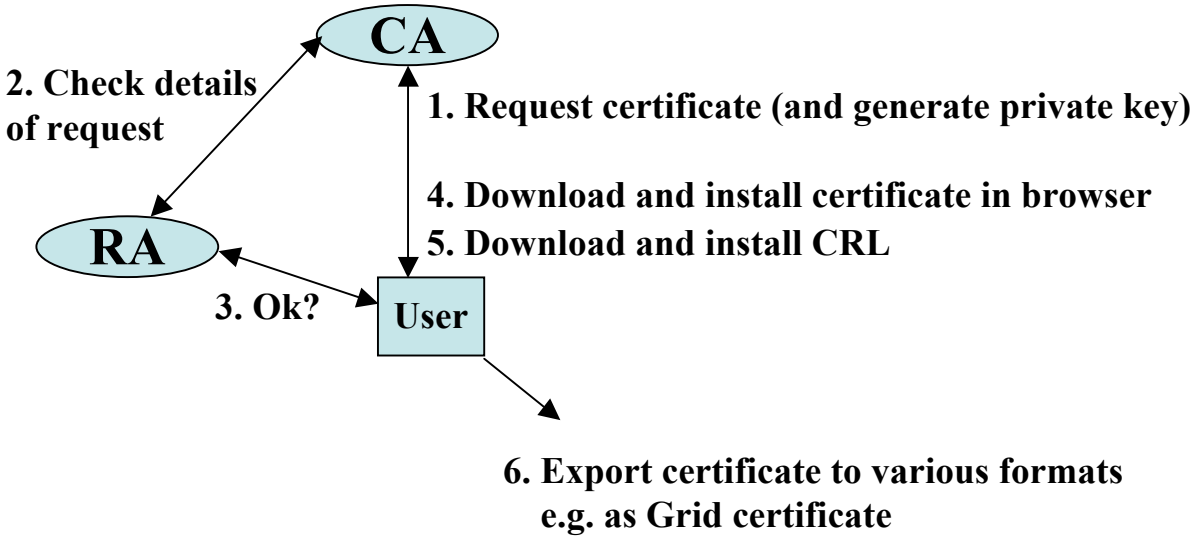
- Other more complex architectures exist
  - Mesh PKIs
    - Trust relationships established on peer-peer basis
      - Webs of trust
        - » Users trust single CA, but trusted CA not same for all users
        - » Certificate path establishment harder – loops possible!



- » Can have hybrid combinations of these (mesh/hierarchical)

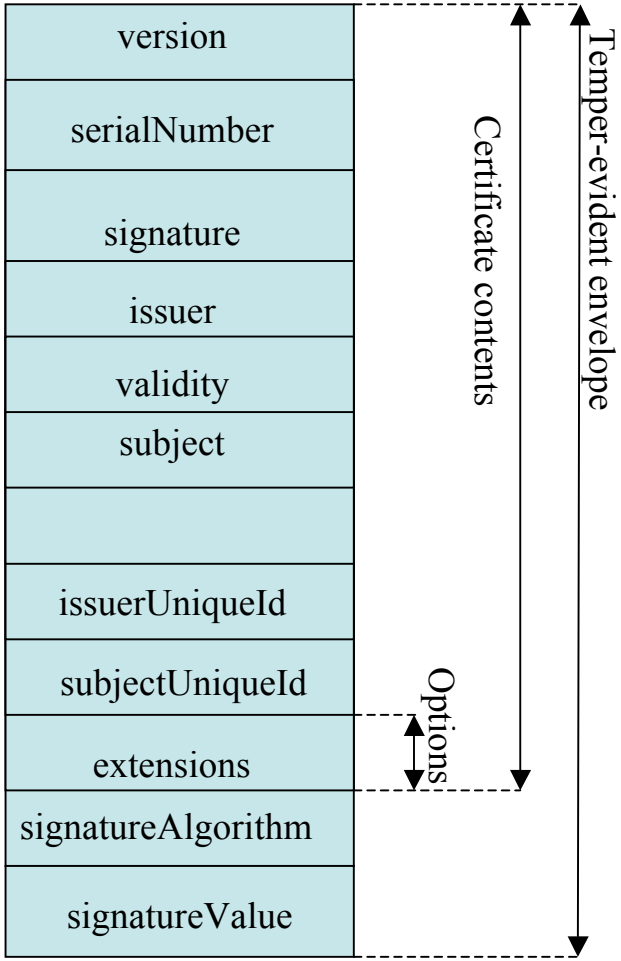
UK e-Science PKI

- Based on statically defined centralised CA with direct single hierarchy to users
- Typical scenario for getting Grid certificate

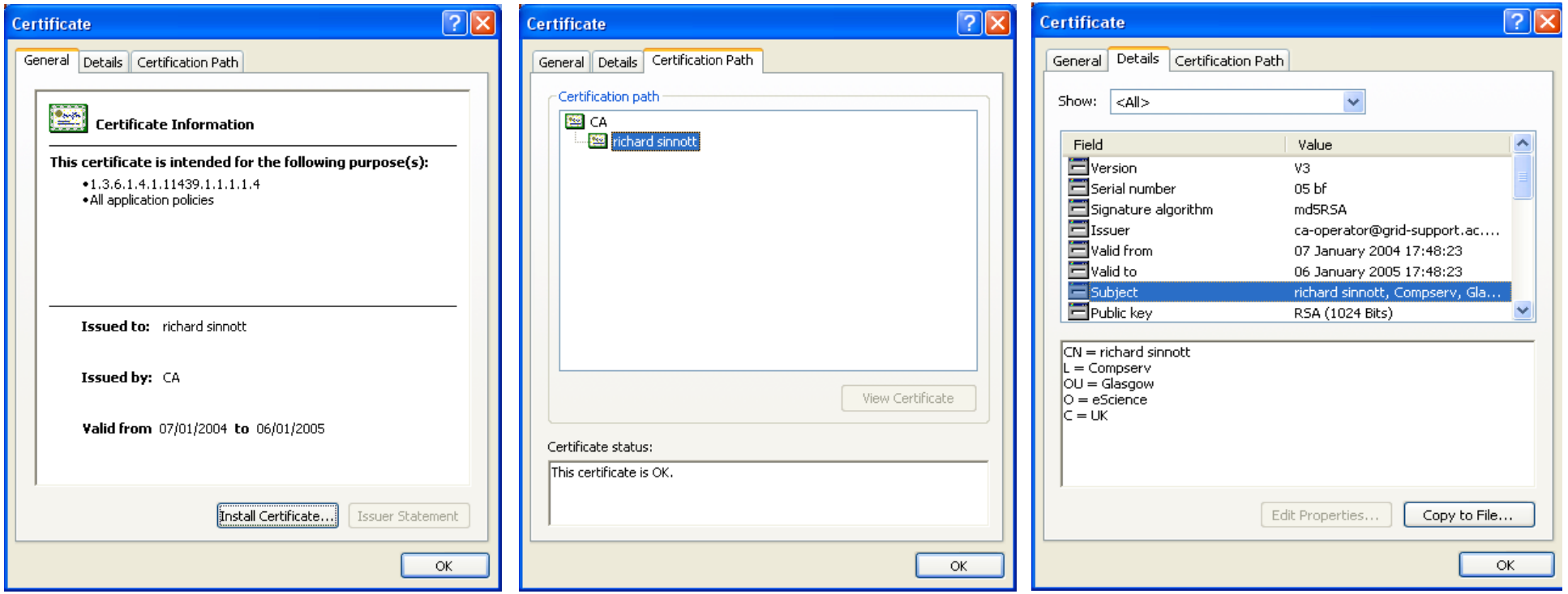


X.509 Certificates

- Consider as 3 main components
  - Tamper-evident envelope
    - So can read certificate without modifying contents
      - <certificate X signatureAlgorithm X signatureValue>
  - Certificate contents
    - Version
      - 1..3, 3= extensions included
    - serialNumber
      - Number given by issuer, unique for each issuer cert.
        - » Useful for CRLs
    - Signature
      - Algorithm identifier, e.g. XXX
    - Issuer
      - X.500 distinguished name (DN)
    - Validity
      - From-to dates/times
        - » 07 January 2004 17:48:23, 06 January 2005 17:48:23
    - Subject
      - X.500 distinguished name for holder of private key
        - » e.g. CN = joe bloggs; L = Computing; OU = NeSC; O = GU; C = UK
    - subjectPublicKeyInfo
      - Subjects public key & algorithm identifier
  - Optional extensions
    - e.g. subjectType (end user, CA…?)



Example X.509 Certificate



- Note on Certificate Policies
  - UK e-Science Certificates are suitable for the following applications:
    - SSL or GSI client (all certificates);
    - SSL or GSI server (server and service certificates only);
    - GSI service (service certificates only);
    - Generating GSI proxies (all certificates);
  - Permissible to use certificates for email signing

PKI Issues

- So what is wrong with PKI
  - Only authentication support (not authorisation)
    - Not able to restrict user actions
    - Collections of users identified and statically defined trust relationships
    - But what if want to dynamically establish a VO where different users have different roles, different responsibilities and resources themselves are changing…?
      - PKIs in themselves do not support this possibility
      - Note that PKIs do support *policies*, e.g. on acceptable usage of PKI resources
        - » Called Certificate Policy, Certificate Policy Statement
      - But in general this is not a directly enforceable contract
        - » e.g. Not at level of "user X can/(cannot) launch job Y on resource Z"
        - » Rather - given more for general high level policy guidelines and information purposes
  - And need all other security aspects (auditing, privacy, …)