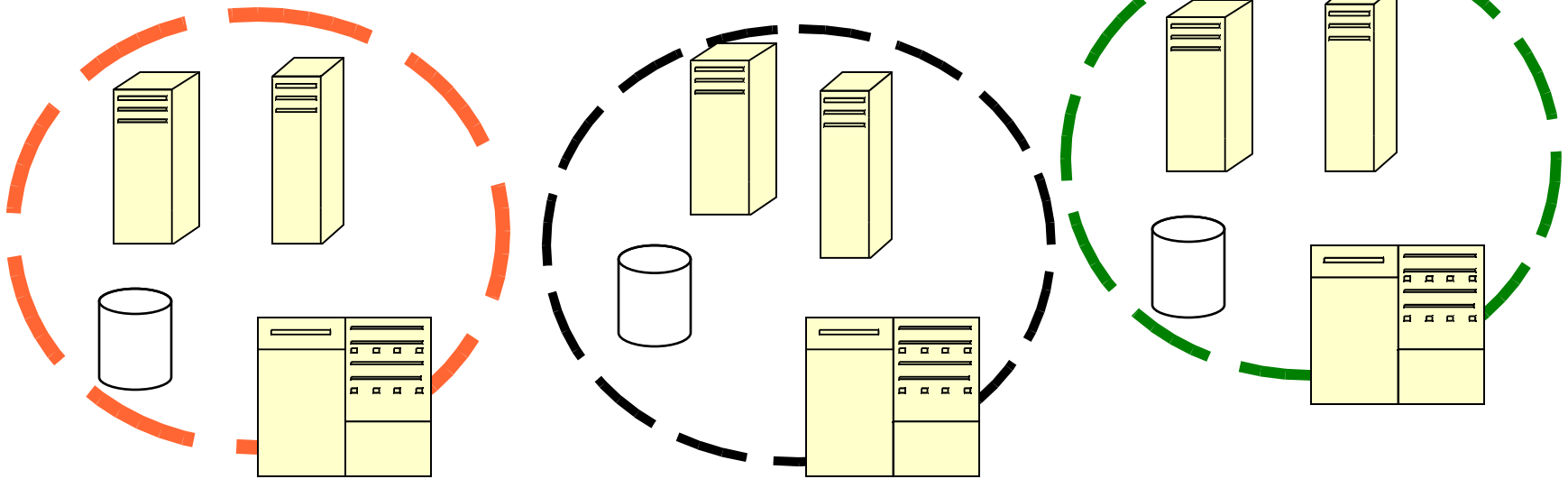


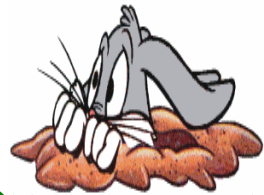
Grid School Module 4: Grid Security

Typical Grid Scenario



Resources

Users



Requirements

The users want to be able to **communicate securely**

Three fundamental concepts (more on the following slides):

- **Privacy** - only invited to understand conversation (use encryption)
- **Integrity** - message unchanged (use signed messages)
- **Authentication** - verify entities are who they claim to be (use certificates and CAs)

One more, very important for grids

- **Authorization** - allowing or denying access to services based on policies.



What do we need ?

- Identity
- Authentication
- Message Protection
- Authorization
- Single Sign On

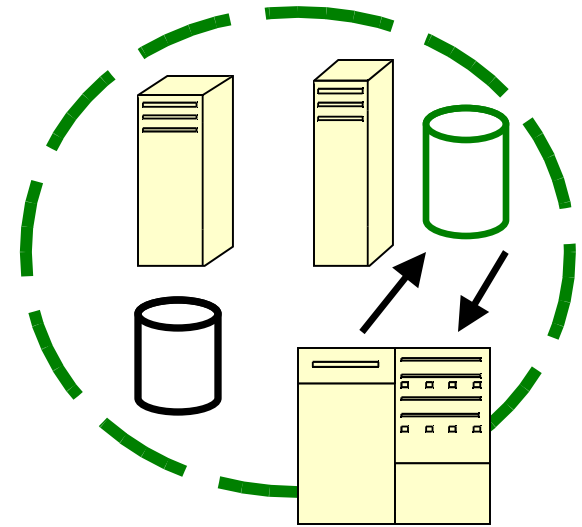
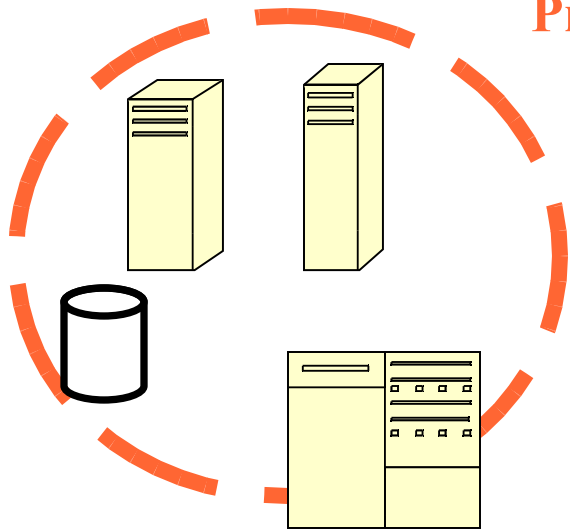


Identity & Authentication

- Each entity should have an identity
- Authenticate: Establish identity
 - Is the entity who he claims he is ?
 - Examples:
 - > Driving License
 - > Username/password
- Stops masquerading imposters



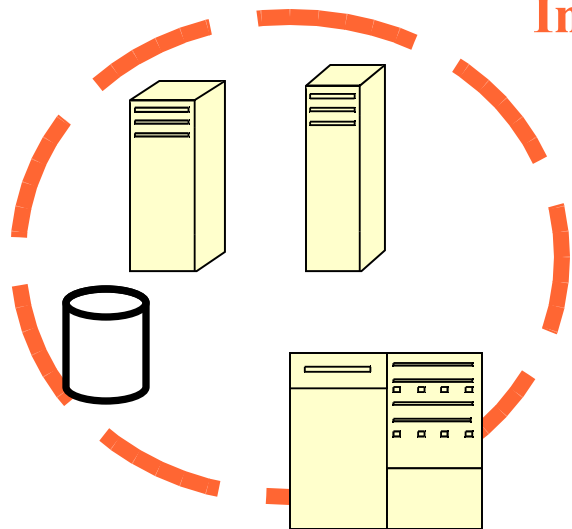
Message Protection: Privacy



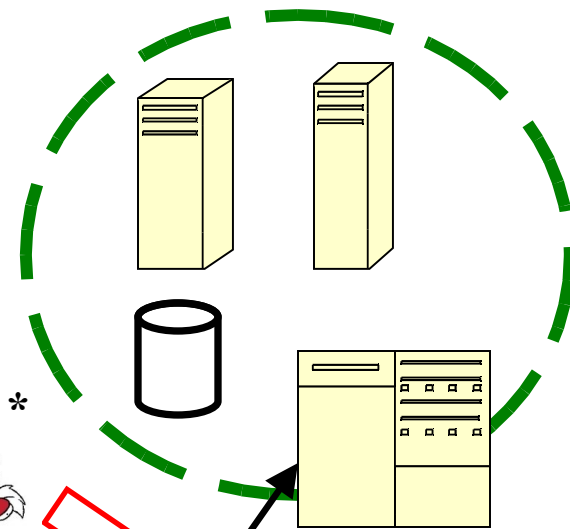
Medical Record
Patient no: 3456



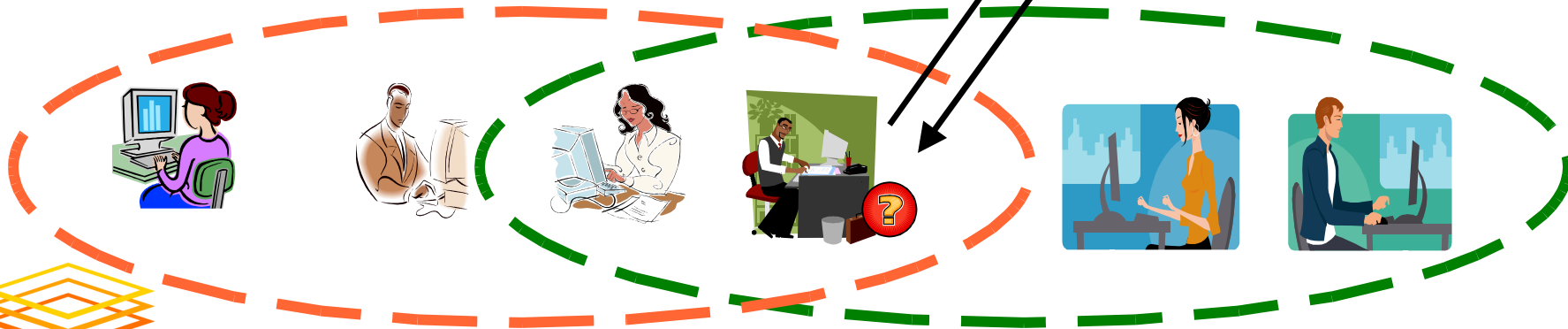
Message Protection: Integrity



Run myHome/rm -f *



Run myHome/whoami



Authorization

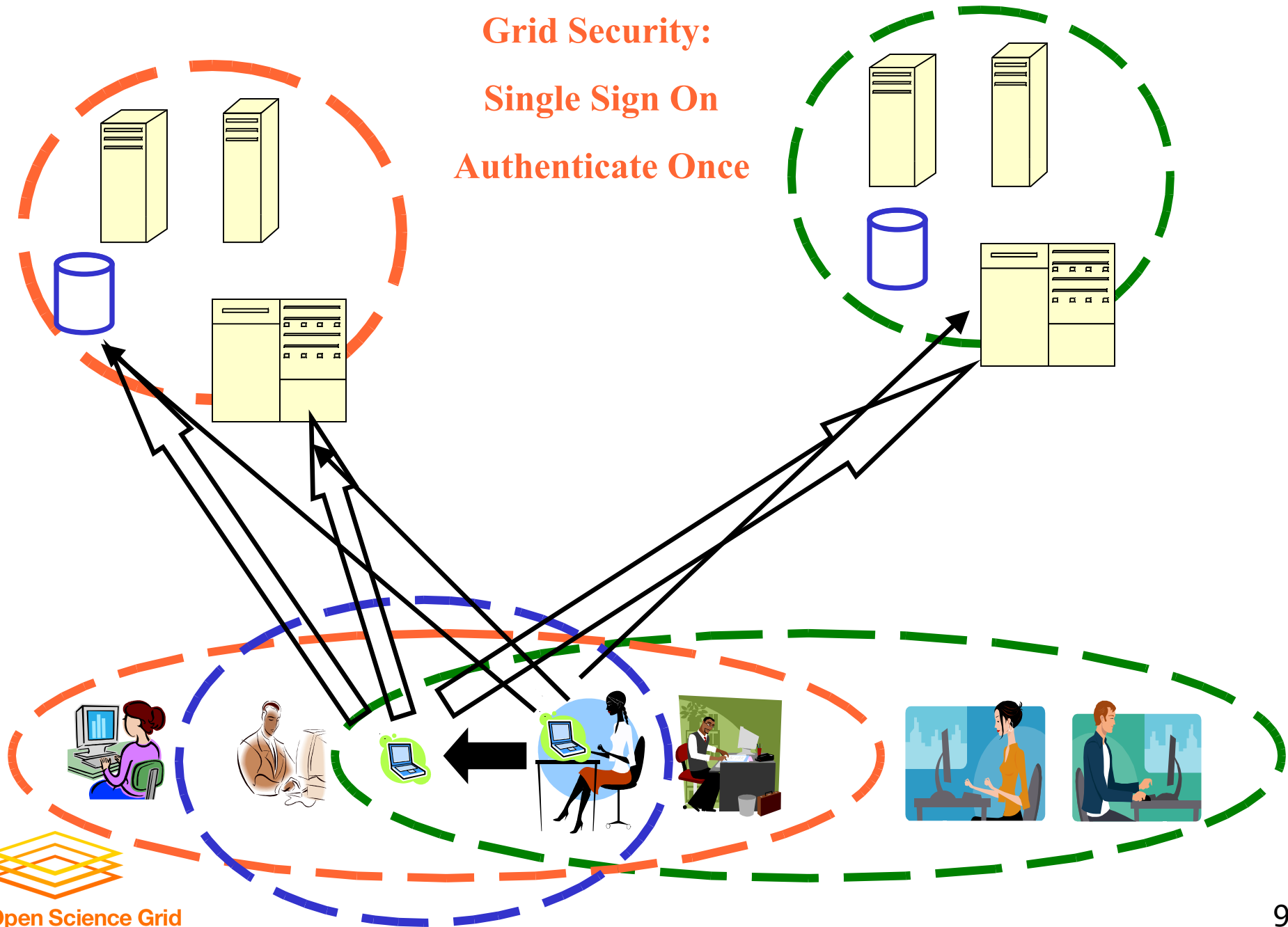
- Establishing rights
- What can a said identity do ?

Examples:

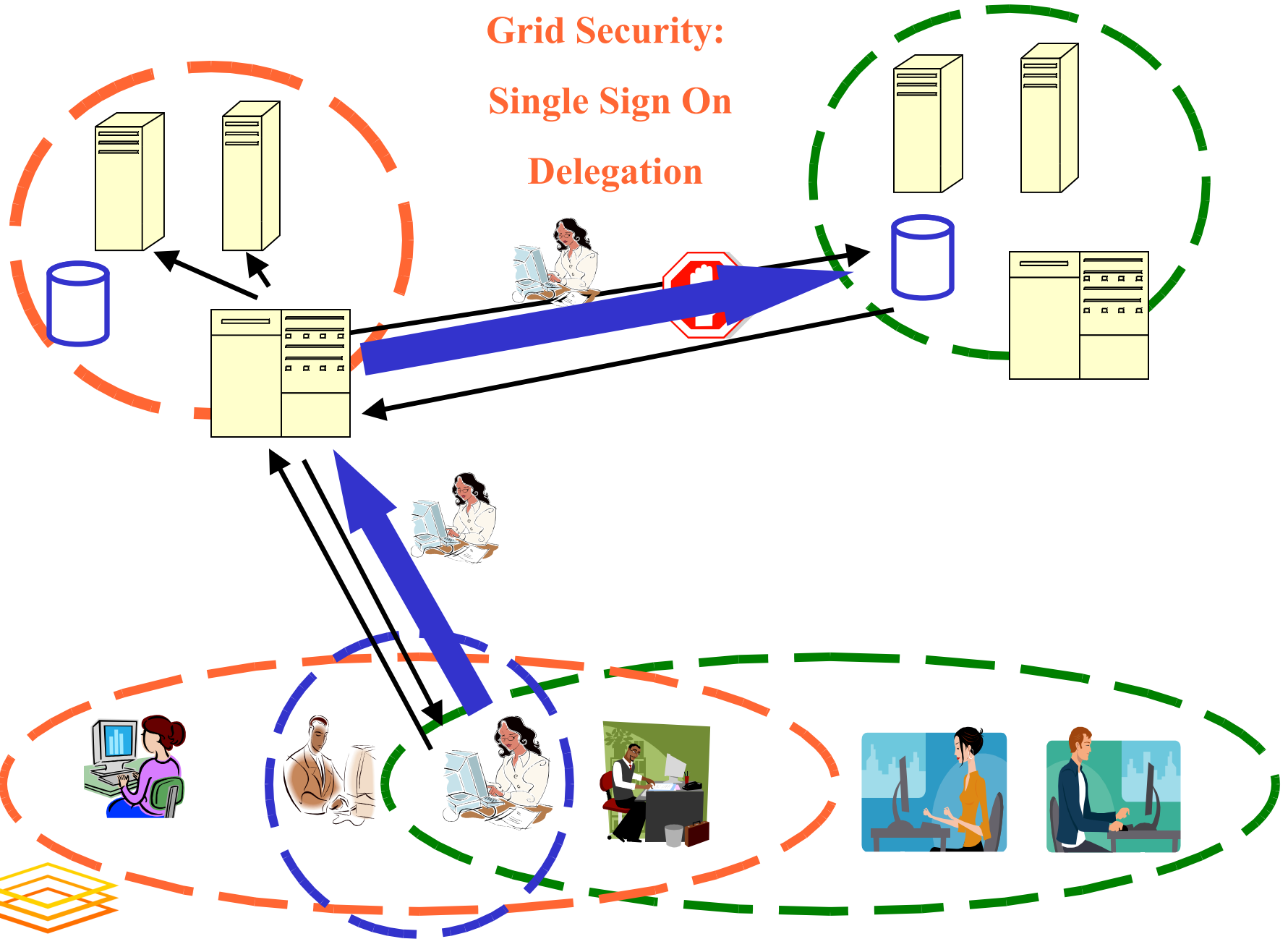
- Are you allowed to be on this flight ?
 - > Passenger ?
 - > Pilot ?
- Unix read/write/execute permissions
- Must authenticate first



**Grid Security:
Single Sign On
Authenticate Once**



Grid Security: Single Sign On Delegation



Single Sign-on

- Important for complex applications that need to use Grid resources
 - Enables easy coordination of varied resources
 - Enables automation of process
 - Allows remote processes and resources to act on user's behalf
 - Authentication and Delegation



Solutions

Issues

- **Resources may be valuable & the problems being solved sensitive**
 - Both users and resources need to be careful
- **Resources and users are often located in distinct administrative domains**
 - Can't assume cross-organizational trust agreements
 - Different mechanisms & credentials
- **Dynamic formation and management of communities (VOs)**
 - Large, dynamic, unpredictable, self-managed ...
- **Interactions are not just client/server, but service-to-service on behalf of the user**
 - Requires delegation of rights by user to service
- **Policy from sites, VO, users need to be combined**
 - Varying formats
- **Want to hide as much as possible from applications!**



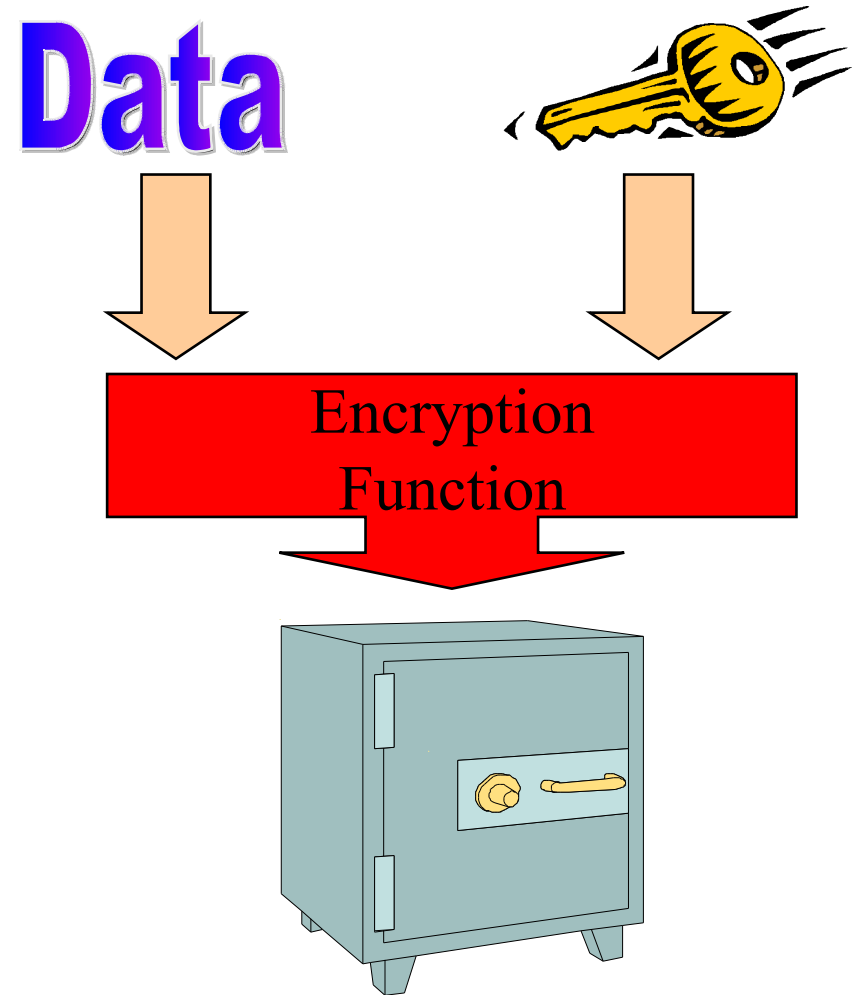
Cryptography for Message Protection

- Enciphering and deciphering of messages in secret code
- Key
 - Collection of bits
 - Building block of cryptography
 - More bits, the stronger the key



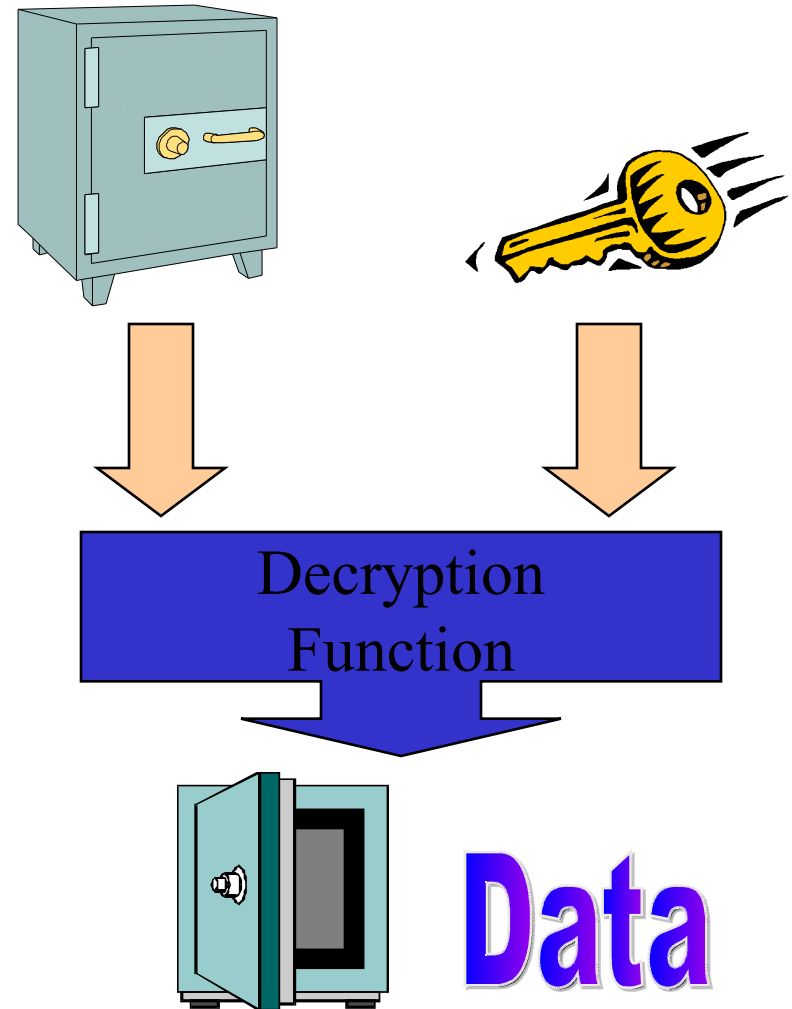
Encryption

- Encryption is the process of taking some data and a key and feeding it into a function and getting encrypted data out
- Encrypted data is, in principal, unreadable unless decrypted



Decryption

- Decryption is the process of taking encrypted data and a key and feeding it into a function and getting out the original data
 - Encryption and decryption functions are linked



Asymmetric Encryption

- Encryption and decryption functions that use a key pair are called asymmetric
 - Keys are mathematically linked



Public and Private Keys

- With asymmetric encryption each user can be assigned a key pair:

a **private** and a **public** key



Private key is known only to owner



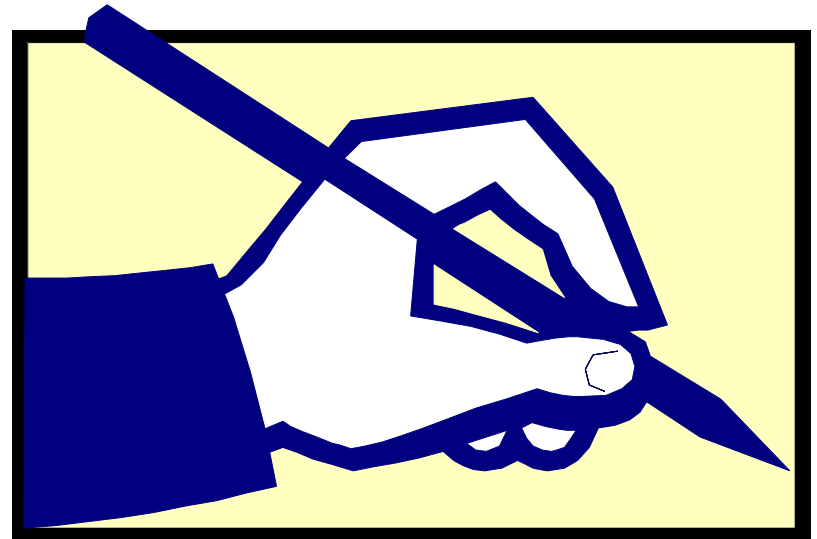
Public key is given away to the world

- Encrypt with public key, can decrypt with only private key
- Message Privacy



Digital Signatures

- Digital signatures allow the world to
 - determine if the data has been tampered
 - verify who created a chunk of data
- Sign with private key, verify with public key
- Message Integrity



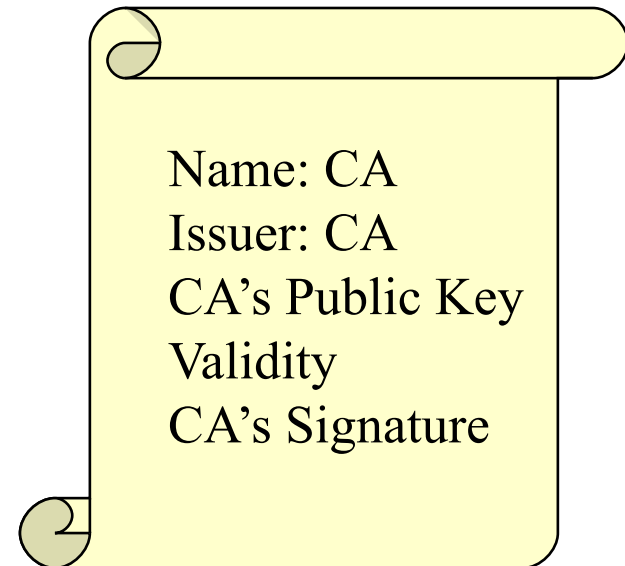
Public Key Infrastructure (PKI)

- PKI allows you to know that a given public key belongs to a given user
- PKI builds off of asymmetric encryption:
 - Each entity has two keys: public and private
 - The private key is known only to the entity
- The public key is given to the world encapsulated in a X.509 certificate



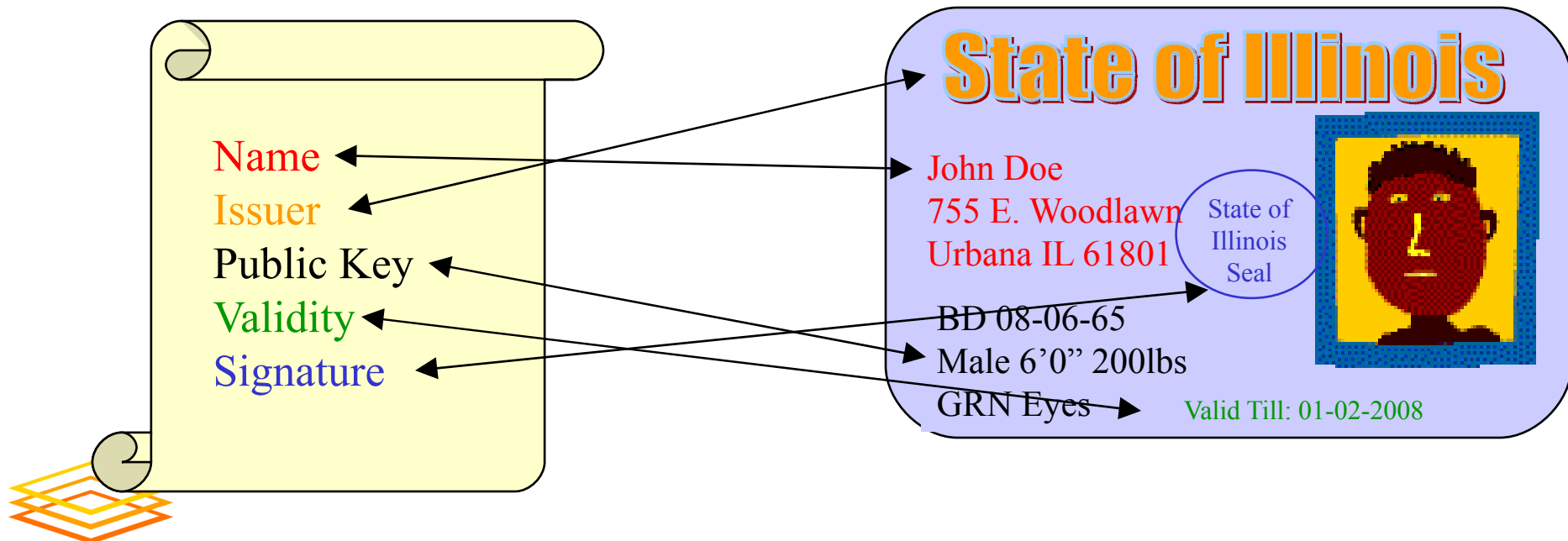
Certification Authorities (CAs)

- A Certification Authority is an entity that exists only to sign user certificates
- The CA signs its own certificate which is distributed in a trusted manner
- Verify CA certificate, then verify issued certificate



Certificates

- X509 Certificate binds a public key to a name.
- Similar to passport or driver's license



Many CA's exist

- Indeed, many CA providers exist
- ESNet:
 - **DOEGrids (doegrids.org)**
 - ESNet Root
 - NorduGrid
 - Russian Data Intensive Grid

In the exercises of this online tutorial,
you will be using DOEGrids



Certificate Policy (CP)

- Each CA has a Certificate Policy (CP) which states
 - who it will issue certificates to
 - how it identifies people to issue certificates to
- Lenient CAs don't pose security threat, since resources determine the CAs they trust.



Certificate Issuance

- User generates public key and private key
 - In our case, through the cert-request command you'll see later on (there are some other tools)
- CA vets user identity using CA Policy
- Public key is sent to CA
 - Email
 - Browser upload
 - Implied
- The CA signs user's public key as X509 Certificate
- User private key is never seen by anyone, including the CA



Certificate Revocation

- CA can revoke any user certificate
 - Private key compromised
 - Malicious user
- Certificate Revocation List (CRL)
 - List of X509 Certificates revoked
 - Published, typically on CA web site.
- Before accepting certificate, resource must check CRLs



Authorization

- Establishing rights of an identity
- Chaining authorization schemes
- Types:
 - Server side authorization
 - Client side authorization



Gridmap Authorization

- Commonly used in Globus for server side authorization
- **Gridmap** is a list of mappings from allowed DNs to user name
 - "/C=US/O=Globus/O=ANL/OU=MCS/CN=Ben Clifford" benc
 - "/C=US/O=Globus/O=ANL/OU=MCS/CN=MikeWilde" wilde(in /etc/grid-security/grid-mapfile directory)
- Controlled by administrator
- Open read access



Globus Security: The Grid Security Infrastructure

- The Grid Security Infrastructure (GSI) is a set of tools, libraries and protocols used in Globus to allow **users** and **applications** to securely access resources.
- Based on PKI
- Uses Secure Socket Layer for authentication and message protection
 - Encryption
 - Signature
- Adds features needed for Single-Sign On
 - Proxy Credentials
 - Delegation



GSI: Credentials

- In the GSI system each user has a set of credentials they use to prove their identity on the grid
 - Consists of a X509 certificate and private key
- Long-term private key is kept encrypted with a pass phrase
 - Good for security, inconvenient for repeated usage
 - Do not use this phrase !

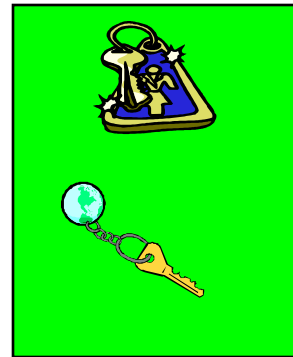
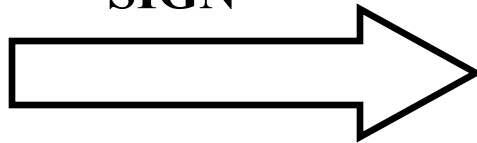


GSI: Proxy Credentials

- Proxy credentials are short-lived credentials created by user
 - Proxy signed by certificate private key
- Short term binding of user's identity to alternate private key
- Same effective identity as certificate



SIGN



GSI: Proxy Credentials

- Stored unencrypted for easy repeated access
- Chain of trust
 - Trust CA -> Trust User Certificate -> Trust Proxy
- Key aspects:
 - Generate proxies with **short** lifetime Set appropriate permissions on proxy file
 - Destroy when done



GSI Delegation

- Enabling another entity to run *as you*
- Provide the other entity with a proxy
- Ensure
 - Limited lifetime
 - Limited capability



Grid Security At Work

- Get certificate from relevant CA
 - DOEGrids in our case
- Request to be authorized for resources
 - Meaning you will be added to the OSGEDU VOMS
- Generate proxy as needed
 - Using grid-proxy-init
- Run clients
 - Authenticate
 - Authorize
 - Delegate as required

Numerous resources, different CAs, numerous credentials



Acknowledgments:

Rachana Ananthakrishnan, Frank Siebenlist,
Von Welch, Ben Clifford - ANL
Åke Edlund, EGEE

