National e-Science Centre

Tutorial 3: Globus Security

Main Points

National e-Science Centre

- Certificates and Proxies (Authentication)
  - X509 Certificates
  - Certificate Authorities
- Access Control Lists (Authorisation)
  - grid-mapfile
- Securing Grid Services
- Beyond GSI…

Security Ingredients

National e-Science Centre

- **Authentication**
  - checking you are who you say you are
- **Authorisation**
  - allowing you to do only what you're allowed to do
- **Audit/accounting**
  - checking what you're doing and when
- **Confidentiality**
  - making the data you use secure
- **Privacy**
  - making sure the data is subsequently used securely
- **Integrity**
  - making sure your data isn't corrupted
- **Fabric management**
  - ensures minimal impact from other applications' security faults
- **Trust**
  - how much we expect you to do what we asked with what we gave you

Options?

National e-Science Centre

- **Kerberos**
  - **Authenticates users through a secure transaction with a centrally maintained key server**
  - **Designates trustworthy key servers in other organisations**
    - Mechanism for inter-organisational authentication
- **OK, but**
  - **Sites need to negotiate many cross-realm authentication agreements**
    - Surrenders too much control of the local security policy.
    - Requires organisation to be completely ‘Kerberos’
      - Inter-site AND intra-site communication

Options?

National e-Science Centre

- **Secure Shell (SSH)**
  - Widely used, PKI based, simple technology
  - Protects user credentials with link encryption
  - Very easily deployed
- **Easy, but**
  - Users need to manage lots of different passwords/public keys for inter-site work
  - No authorisation control
    - Without invasion of privacy
  - Only supports simple tasks (remote shell or file transfer)
    - Doesn't support collaborative environments (or Web browsers!!!)

Grid Security Infrastructure

National e-Science Centre

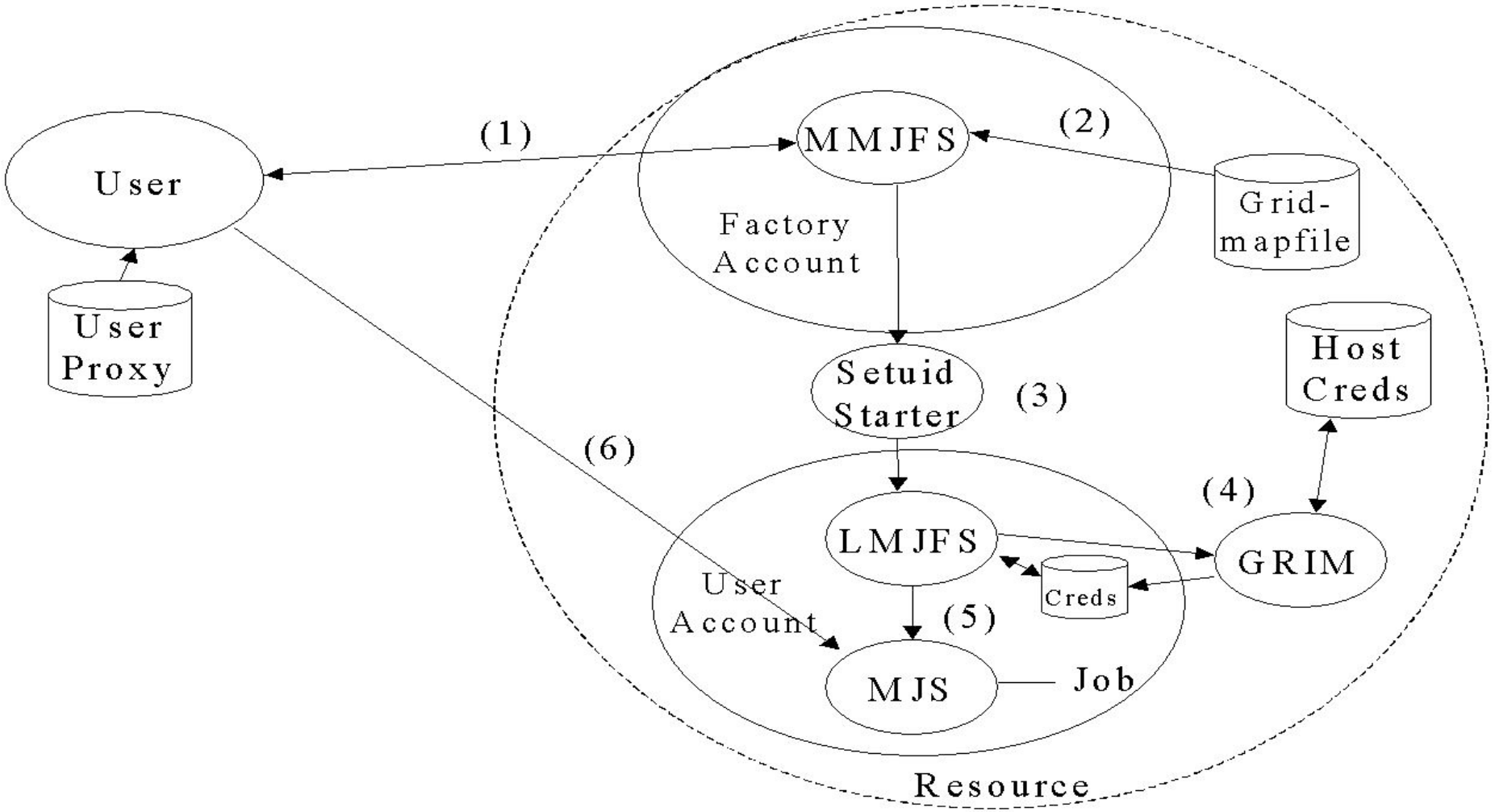
- The implementation of secure functionality in Globus Toolkit…
  - Provides secure communication between grid elements
  - Preserves site control for access policies and local security
  - Supports “single sign-on” and delegation of credentials for applications that require multiple resources
- All elements of the Globus Toolkit are built on top of this basic infrastructure

Security

National e-Science Centre

- **Will make (and/or) break your Grid Services!**
  - Expect to spend most of your time securing your services
  - Tends to be very user-unfriendly, requires modification of fundamental globus files such as the container deployment descriptor
  - Standards in Authz haven't been agreed on yet
    - c.f. OGSA vs WSRF – changing standards, but standards nonetheless!
  - Expect some ad-hoc solutions to problems!

Recap of Job Submission



Step 1

National e-Science Centre

- Client generates a proxy user certificate
  - Using grid-proxy-init
- Client generates a job request
  - (globusrun equivalent)
- Client signs this request with their proxy credentials and sends to the Master Managed Job Factory Service (MMJFS) on the remote resource.

Step 2

- The MMJFS runs in a non-privileged factory account.
  - e.g. user ‘globus’ in a typical GT3 install
- It verifies the signature on the request and establishes the identity of the user who sent it.
- It then determines the local account in which the job should be run.
  - Currently this is done by using the grid-mapfile and user's grid identity.

Step 3

- **The MMJFS invokes the *setuid starter* process to start a Local Managed Job Factory Service (LMJFS) in the user's account**
  - **Assuming the user does not already have one running in their account**

- ***setuid starter***
  - **a small setuid program running with root privileges that has the SOLE function of starting the LMJFS in the user's account.**

Security!!

Step 4

National e-Science Centre

- LMJFS calls the Grid Resource Identity Mapper (*GRIM*) to acquire a set of credentials.
- This ‘proxy’ credential has embedded in it
  - the user's Grid identity
  - the local account name and
  - local policy about the user.
    - The latter policy is obtained from the Grid map file entries that apply to that local account.
- *GRIM* is a setuid program that accesses the local host credentials and from them generates the proxy for the LMJFS.

Step 5

National e-Science Centre

- **MMJFS then forwards the original user-signed job instantiation request from the user to the LMJFS.**
  - **LMJFS verifies the signature on the request**
    - make sure it has not been tampered with and to make sure it was created by a user that is authorized to run in the local user account.
- **Once these checks are successfully completed, the LMJFS instantiates a Managed Job Service (MJS), presents it with the user's request, and returns a reference to the MJS to the user**

Step 6

National e-Science Centre

- User connects to the MJS.
- User and MJS then perform mutual authentication
  - the user using their proxy and the MJS using the credentials acquired from *GRIM*.
- The MJS authorizes the user as a valid user to access the local account it is running in.
  - The user authorizes the MJS as having a credential issued from an appropriate host credential and containing a Grid identity matching it's own, thus verifying the MJS it is talking to is running not only on the right host, but in an appropriate account. The user would then delegate GSI credentials to the MJS for the job to use and start the job running.

Certificates

National e-Science Centre

- **Your 'login' to the Grid:**
  - **X509 Format – web browser compatible**
  - **Issued by a Certificate Authority**
    - International (e.g. the O=Grid/OU=Globus test CA)
    - National scale (e.g. the UK e-Science Certificate Authority)
    - Lab scale (the NeSC Lab simpleCA, or the PERMIS PMI)
  - **Usually contains encrypted private key unlocked by password…**
    - This is the "weakest link"
    - Key protected locally by permissions
  - **Allows single-sign-on (SSO) through *proxies***

GSI certificates

- **GSI certificates includes four primary pieces of information**
  - **A "subject"**
    - Identifies the person or object the certificate represents
  - **A public key belonging to the subject**
  - **The identity of a third party (the Certificate Authority) that has signed the certificate to certify the public key and subject name belong to the subject**
    - To trust the certificate you MUST trust the CA
  - **The digital signature of the CA**
    - To ensure the information on the certificate hasn't been changed

Delegation and single sign-on

National e-Science Centre

- **Essential to easy access of multiple resources**
  - **GSI extends the SSL protocol to reduce the number of times you have to enter your passphrase**
  - **Done through the creation of *proxy certificates***
    - A proxy consists of a new *short-lifetime* certificate (with a new public key within it) and a new private key
    - The certificate is signed by the OWNER not the CA
    - The new private key must be kept secure – but since it is not valid for long it is ok to store it locally unencrypted – the file permissions should be enough
    - This proxy certificate and private key may be used for mutual authentication without the need to enter a password
      - The process differs slightly but establishes a chain of trust from the CA, through the owner, to the created proxy

A “typical” grid-mapfile

National e-Science Centre

```
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=steve kee" skee
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=stewart mills" smills
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=iain mcbride" imcbride
"/C=UK/O=eScience/OU=Aberdeen/L=GeSC/CN=nikki salter" nsalter
"/C=UK/O=eScience/OU=Newcastle/L=NEReSC/CN=nicola wightman" nwightman
"/C=UK/O=eScience/OU=London/L=LeSC/CN=scott mccaig" smccaig
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=kev mcneil" kmcneil
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=nik martin" nmartin
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=ann robertson" aroberts
```

- **Format:**
  - **“<Grid DN of user>”<space><local account name>…**
    - DN must be in quotes as the surname tends to get chopped off if you miss it out

Certificate Tools

National e-Science Centre

- `$ grid-proxy-init`
  - Creates a proxy certificate based on the certificate obtained from `~/.globus`
- `$ grid-cert-info`
  - Returns information about the certificate a user has
  - Passing `-subject` switch returns the DN of the user
- `$ grid-proxy-info`
  - Returns information about a user's proxy certificate
  - `-f /tmp/x509up_u$UID` switch will give you the proxy details

Location…

- **/etc/grid-security**
  - **GSI's main directory**
  - **Contains**
    - hostcert.pem - the resource grid certificate
    - hostkey.pem - its corresponding private key
    - grid-mapfile - the mapping of identity (DN) to local user account
    - grim-port-type - the mapping of user account to valid portTypes
- **/etc/grid-security/certificates**
  - **GSI's CA information**
    - *ca_hash*.0 - the root certificate of the CA (trust anchor)
    - *ca_hash*.signing_policy – the scope of the CA

Location…

National e-Science Centre

- **~/.globus**
  - **GSI directory for user information**
  - **Contains:**
    - usercert.pem – the user grid certificate
    - userkey.pem – the private key for the user certificate
  - **Certificates should be read-only for *all***
  - **Keys should be read-only for *user***
    - GSI sensitive to the permissions on these certificates
      - grid-proxy-init fails if user has write permission on certificate!
      - Any other key permission will fail.

Location.

National e-Science Centre

- /tmp/
  - GSI uses this directory to store proxy certificates
  - Contains
    - x509cp_globus_grim – the container proxy certificate
    - x509up_u$UID - the user proxy certificate
  - $ grid-proxy-info –f /tmp/x509up_u$UID
    - Contains three sections
      - A new proxy certificate
      - A private key for this proxy
      - The issuing certificate
    - NB. The issuing certificate may have been another proxy

Securing a Service

National e-Science Centre

- **Writing a secure service requires two changes to the infrastructure**
  - 1. Add appropriate security parameters to your container deployment descriptor to activate GSI security
  - 2. Insert GSI stubs to the client class to enable secure conversation between client/host
  - N.B. This assumes you have an operational Grid Service (with client) already deployed.

Service side

National e-Science Centre

- **Identify your service in the container deployment descriptor (`$GLOBUS_LOCATION/server-config.wsdd`)**
  - **Add two parameter tags to your service description**
  - `<parameter name="authorization" value="*****"/>`
    - Where ***** could be 'none','self','gridmap' or 'custom'
  - `<parameter name="securityconfig" value="org/globus/ogsa/impl/security/descriptor/gsi-security-config.xml"/>`
    - Specifies all the methods we need for secure conversation

Client side

National e-Science Centre

- **Need to add two lines to the client class**
  - `((Stub)service)._setProperty(Constants.GSI_SEC_CONV,Constants.ENCRYPTION);`
  - `((Stub)service)._setProperty(Constants,AUTHORIZATION,NoAuthorization.getInstance());`
  - **First line defines what security level to use**
    - Mostly full-blown ENCRYPTION or digital SIGNATURE
  - **Second line defines client side authZ**
    - Not usually needed, hence “NoAuthorization”
      - Usually get server side to perform authZ

Custom Security

National e-Science Centre

- GSI is a generic framework for Grid Security
- However it has allowed for custom applications to do some of its work
  - e.g. Authentication may be done through a portal or through an existing Novell-style login
    - See Portals tutorial Week 8
    - Still need underlying GSI certificates for Grid job submission
- Authorization is very coarse in GSI
  - GGF specify a SAML callout function which allows a third-party app to make AuthZ decisions
    - PERMIS…

Finally…

- **Credential check**
  - The user certificates you have been using were created using the Globus simpleCA.
  - `grid-cert-info -subject` should return a string like:

`"/O=Grid/OU=Globus/OU=simpleCA-labpc-1.nesc.gla.ac.uk/OU=nesc.gla.ac.uk/CN=your name"`

  - These certificates CANNOT be used with PERMIS (one of the tools for the programming exercise)
  - We have created special certificates which are compatible with all the tools you will need

Finally…

- In your ~/.globus directory, there will be a subdirectory called permis/
  - These credentials (usercert and userkey) must be placed into .globus when you wish to test your assignment code.
  - Remember to copy them into /home/globus/.globus/
  - You won't need the simpleCA credentials any more once you have started your assignment
  - The DN will look like
  - `"/C=gb/O=glasgow/OU=dcs/CN=your name"`