

Grid Security Tutorial

Grid Computing module

20th February 2007



University of Glasgow, Scotland

Anthony Stell



UNIVERSITY
of
GLASGOW

Computer Security



- ***“Can a user depend on the software and system behaving in a manner that they expect?”***



“AAA”

- **Authentication**
 - Verifying the identity of an individual...
- **Authorization**
 - Verifying the privilege of that individual based upon their identity...
- **Accountability**
 - Holding an individual to account in the event of a compromise in security...
- **Other aspects of security exist (confidentiality, data integrity, etc.) but these are the basics...**

Books to read before you die...

- ***“Secrets and Lies: Digital Security in a Networked World” - Bruce Schneier***
- ***“The Code Book” – Simon Singh***
- ***“Web Security, Privacy and Commerce” – Simson Garfinkel and Gene Spafford***



Grid Definitions

- *“The Internet is about computers talking to each other; the Grid is about computers working with each other”* – Tom Hawk, IBM
- *“Co-ordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations”* – Ian Foster, Globus Alliance
- The critical point is that you’re trying to “do stuff” across domains that don’t necessarily trust each other...
 - A term that you’ll hear a lot about is a Virtual Organisation (VO): which is a loose collaboration of service or resource providers working together to achieve a common goal.

Authentication on the Grid

- **Difficult.**
- **Could establish a ring-fence around the various resources...**
 - But then you lose flexibility in expanding the VO, it becomes a static collaboration.
 - How do you add in new resources?
 - How rigorous do you make the screening to allow resources to be added to the VO (and hence “within the fenced area”)?
 - How do you establish trust between your site and a remote one with which you have no relationship?
- **Shibboleth goes some way to addressing these issues...**



Authorization on the Grid

- Much, *much* more difficult.
- You've established the identity of a user/client, now you want to enforce an access control policy...
 - But how do you design a generic policy that will cover all possible remote use-cases?
 - Role-Based Access Control (RBAC) makes the problem slightly more manageable, but not completely – you still need to match roles from remote sites to your local policies.
 - How do you manage conflicts of interest?
 - How do you match roles that have no similar classification in your policy (this is the idea of ontologies and data description – see the OGSA-DAI tutorial...)
- No technology has addressed this problem effectively yet...



Security Assertion Markup Language (SAML)

- An XML standard for exchanging security information (mainly authN and authZ assertions) between services and their clients.
 - Fairly well-established protocol...
- OASIS specification can be found here:
 - <http://www.oasis-open.org/specs/> (under SAML v2.0)
 - Good Wikipedia entry tells you more...
- OpenSAML implementation:
 - <http://www.opensaml.org/>

eXtensible Access Control Markup Language (XACML)

- **Similar to SAML but focused on the AuthZ aspects of security**
 - A language to allow the easy description of access control policies.
 - One major benefit is the use of parametric authorization...
- **OASIS Specification can be found here:**
 - <http://www.oasis-open.org/specs/> (under XACML v2.0)
- **Only (?) implementation is by Sun:**
 - <http://sunxacml.sourceforge.net/>

Technologies

- **Lots of “solutions”:**
 - Grid Security Infrastructure (GSI)
 - PERMIS
 - Shibboleth
 - Virtual Organisation Membership Service (VOMS)
 - Akenti
 - Community Authorization Service (CAS)
- **There are issues with all of these because of two major problems:**
 - Software built for something else is being shoe-horned into the Grid technology space
 - Because of this, developers tend to misrepresent the tenets of Grid technology (“Design creep” versus tight deadlines) with the software



GSI

- **Globus implementation of Grid Security**
- **Leverages PKI and OpenSSL to achieve secure transactions between grid services and their clients**
- **Lots of hard-wired constructs:**
 - Huge, complex libraries must be stored on client-side
 - Credentials have to be stored in specific places
 - Programming secure services is complicated and prone to error
 - The grid-map file concept is centralised and not conducive to building scaleable grids...

PERMIS

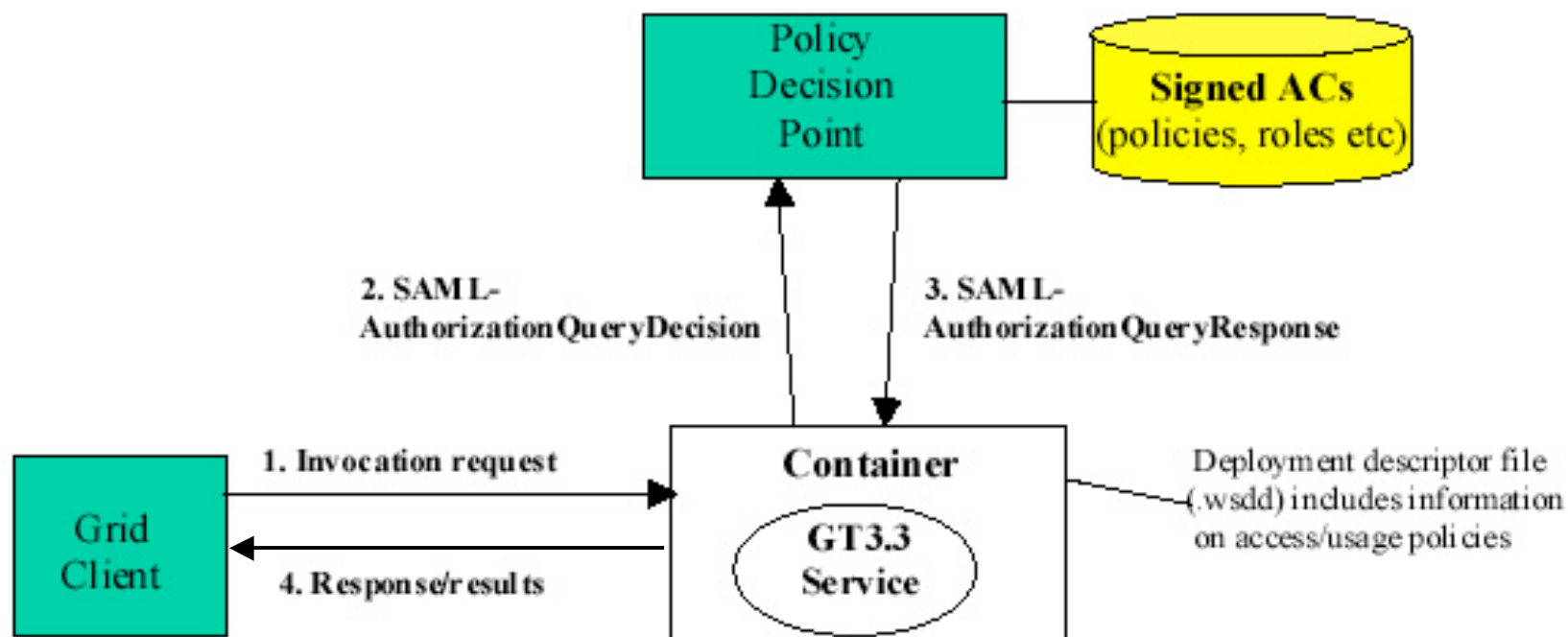
- **“PrivilEge and Role Management Infrastructure and Standards validation”**
- **Essentially a module that interprets SAML**
- **Takes a SAML assertion from a grid service (coded using Globus Toolkit) and compares against a policy.**
- **Returns an authorization decision**
- **Uses “role-based” access control**
- **More than a look-up table – it defines a Privilege Management Infrastructure (PMI)**



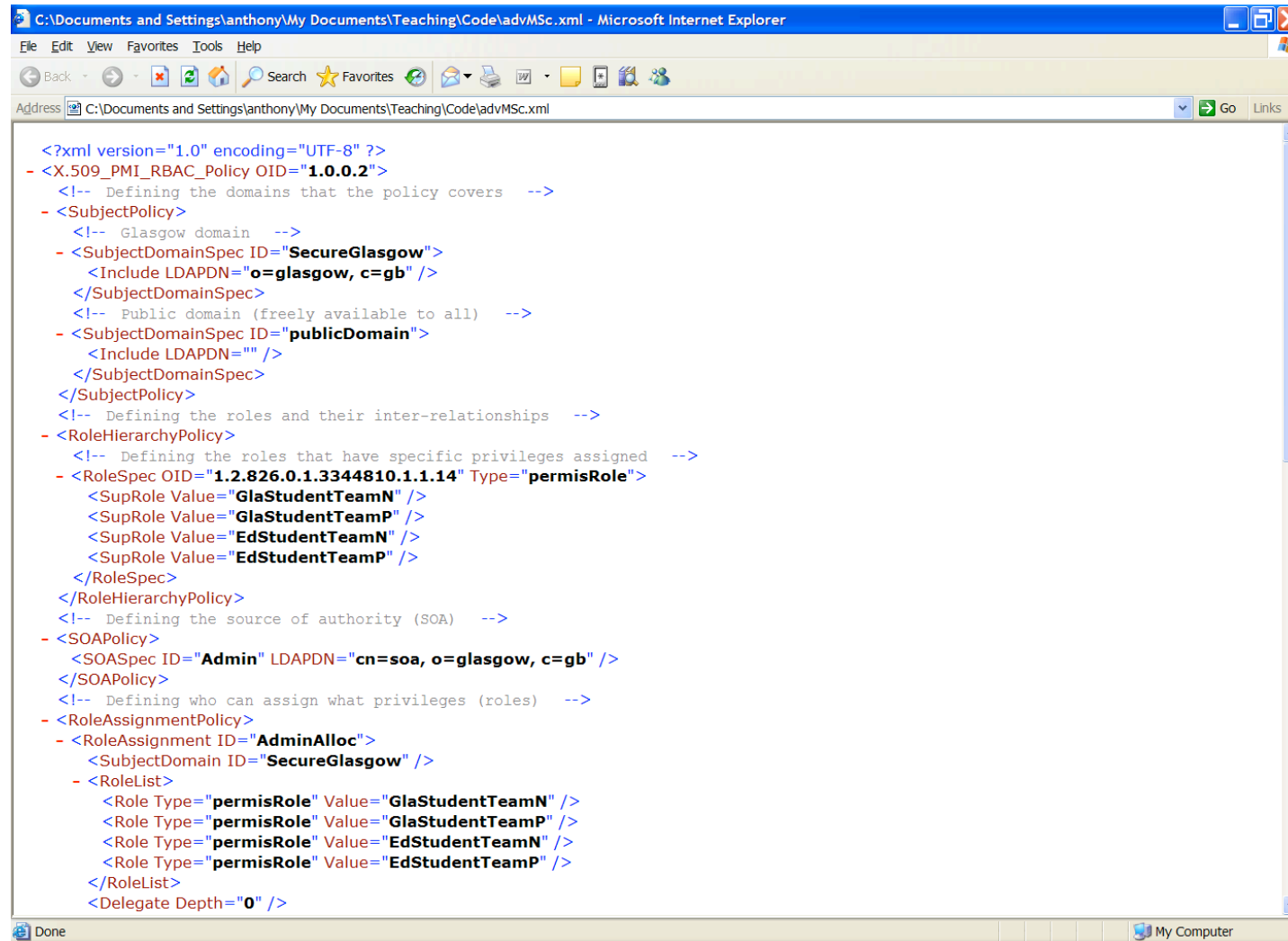
PKI vs. PMI

<u>Concept</u>	<u>PKI Entity</u>	<u>PMI Entity</u>
Certificate	Public Key Certificate (PKC)	Attribute Certificate (AC)
Certificate Issuer	Certification Authority (CA)	Attribute Authority (AA)
Certificate User	Subject	Holder
Certificate Binding	Subject's name to Public Key	Holder's Name to Privilege Attribute(s)
Revocation	Certificate Revocation List (CRL)	Attribute Certificate Revocation List (ACRL)
Root of trust	Root Certification Authority or Trust	Source of Authority (SOA)
Subordinate authority	Anchor Subordinate Certification Authority	Attribute Authority (AA)

PERMIS Architecture



XML Policy



C:\Documents and Settings\anthony\My Documents\Teaching\Code\advMSc.xml - Microsoft Internet Explorer

File Edit View Favorites Tools Help

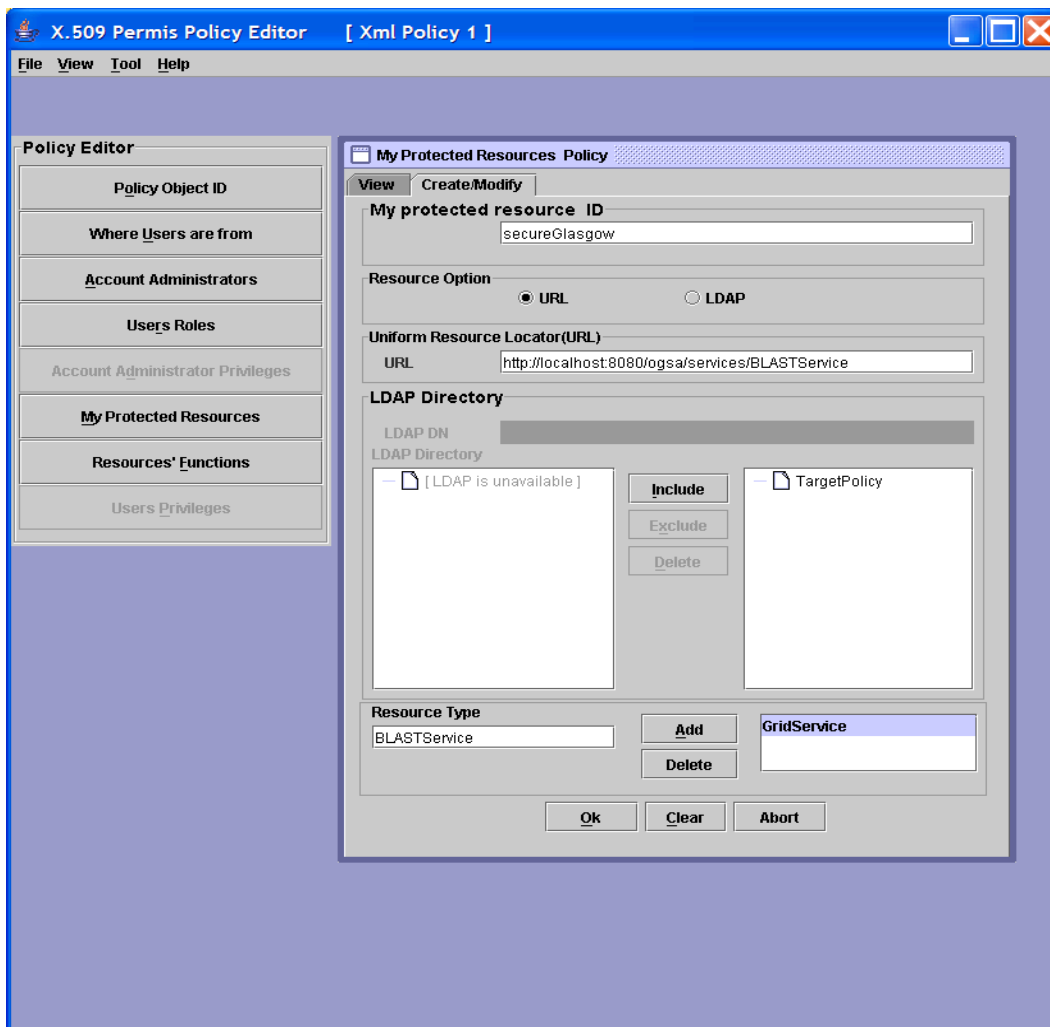
Address C:\Documents and Settings\anthony\My Documents\Teaching\Code\advMSc.xml Go Links

```
<?xml version="1.0" encoding="UTF-8" ?>
- <X.509_PMI_RBAC_Policy OID="1.0.0.2">
  <!-- Defining the domains that the policy covers -->
  - <SubjectPolicy>
    <!-- Glasgow domain -->
    - <SubjectDomainSpec ID="SecureGlasgow">
      <Include LDAPDN="o=glasgow, c=gb" />
    </SubjectDomainSpec>
    <!-- Public domain (freely available to all) -->
    - <SubjectDomainSpec ID="publicDomain">
      <Include LDAPDN="" />
    </SubjectDomainSpec>
  </SubjectPolicy>
  <!-- Defining the roles and their inter-relationships -->
  - <RoleHierarchyPolicy>
    <!-- Defining the roles that have specific privileges assigned -->
    - <RoleSpec OID="1.2.826.0.1.3344810.1.1.14" Type="permisRole">
      <SupRole Value="GlaStudentTeamN" />
      <SupRole Value="GlaStudentTeamP" />
      <SupRole Value="EdStudentTeamN" />
      <SupRole Value="EdStudentTeamP" />
    </RoleSpec>
  </RoleHierarchyPolicy>
  <!-- Defining the source of authority (SOA) -->
  - <SOAPolicy>
    <SOASpec ID="Admin" LDAPDN="cn=soa, o=glasgow, c=gb" />
  </SOAPolicy>
  <!-- Defining who can assign what privileges (roles) -->
  - <RoleAssignmentPolicy>
    - <RoleAssignment ID="AdminAlloc">
      <SubjectDomain ID="SecureGlasgow" />
      - <RoleList>
        <Role Type="permisRole" Value="GlaStudentTeamN" />
        <Role Type="permisRole" Value="GlaStudentTeamP" />
        <Role Type="permisRole" Value="EdStudentTeamN" />
        <Role Type="permisRole" Value="EdStudentTeamP" />
      </RoleList>
      <Delegate Depth="0" />
    </RoleAssignment>
  </RoleAssignmentPolicy>
</X.509_PMI_RBAC_Policy>
</X.509_PMI_RBAC_Policy>
```

Done My Computer



PERMIS Tools: Policy Editor



X.509 Permis Policy Editor [Xml Policy 1]

File View Tool Help

Policy Editor

- Policy Object ID
- Where Users are from
- Account Administrators
- Users Roles
- Account Administrator Privileges
- My Protected Resources**
- Resources' Functions
- Users Privileges

My Protected Resources Policy

View Create/Modify

My protected resource ID
secureGlasgow

Resource Option
☒ URL ☐ LDAP

Uniform Resource Locator(URL)
 URL http://localhost:8080/fgsa/services/BLASTService

LDAP Directory
 LDAP DN
 LDAP Directory

☐ [LDAP is unavailable]

☐ TargetPolicy

Resource Type
 BLASTService

GridService

Ok Clear Abort

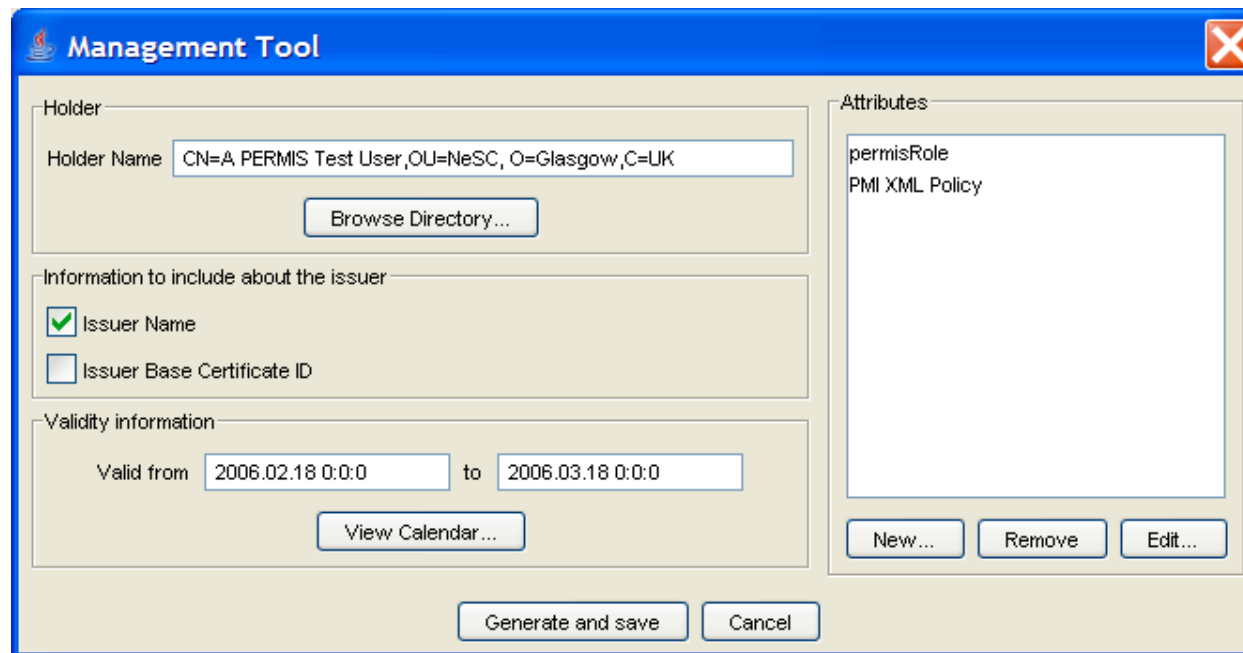
Target Access Policy

```
<TargetAccessPolicy>
  <TargetAccess ID="studentTeamNAccess">
    <RoleList>
      <Role Type="permisRole" Value="GlaStudentTeamN"/>
    </RoleList>
    <TargetList>
      <Target Actions="runBLASTN">
        <TargetDomain ID="BlastService"/>
      </Target>
    </TargetList>
  </TargetAccess>
  <TargetAccess ID="studentTeamPAccess">
    <RoleList>
      <Role Type="permisRole" Value="GlaStudentTeamP"/>
    </RoleList>
    <TargetList>
      <Target Actions="runBLASTP">
        <TargetDomain ID="BlastService"/>
      </Target>
    </TargetList>
  </TargetAccess>
</TargetAccessPolicy>
```



PERMIS Tools: Privilege Allocator

- **Tool to create signed Attribute Certificates:**
 - Standard format, stored in an LDAP directory
 - Can store roles and/or XML policies



The screenshot shows a 'Management Tool' window with the following sections:

- Holder:** A text field for 'Holder Name' containing 'CN=A PERMIS Test User,OU=NeSC,O=Glasgow,C=UK' and a 'Browse Directory...' button.
- Information to include about the issuer:** Two checkboxes: 'Issuer Name' (checked) and 'Issuer Base Certificate ID' (unchecked).
- Validity information:** Two text fields for 'Valid from' (2006.02.18 0:0:0) and 'to' (2006.03.18 0:0:0), with a 'View Calendar...' button.
- Attributes:** A list box containing 'permisRole' and 'PMI XML Policy', with 'New...', 'Remove', and 'Edit...' buttons below it.
- Buttons:** 'Generate and save' and 'Cancel' at the bottom.

Under the hood...

- **LDAP**
 - Back-end directory
- **OpenSSL**
 - Used to set up the PKI certificates that PERMIS necessarily uses.
- **XML**
 - Used to describe policy – nice and hierarchical
- **GSI**
 - Used to secure the calls between service and policy engine



Limitations

- **Method-only execution**
 - My personal bug-bear...
 - Can only run a method, which will then say “yes” or bomb out with an authorization exception
- **Big overhead of supporting infrastructure**
 - Need to appreciate the niceties of Globus, OpenSSL, LDAP...
 - Lots of scope for things to go wrong...
- **Not a mature technology yet**



Shibboleth

- **Attribute exchange mechanism that allows the passing of authentication/authorization assertions between nodes.**
 - Can set up a distributed trust domain...
- **Provides a dynamic single sign-on facility to a “federation” of nodes**
 - This is the most promising step towards establishing a VO so far (imho)...



Virtual Organisations for Trials and Epidemiological Studies (VOTES)

- Bringing Grid technology to clinical trials and the medical domain
 - <http://www.nesc.ac.uk/hub/projects/votes>
- Patient confidentiality, and therefore security, is paramount...
- Have come up with a new Access Control Matrix method of applying privileged authorization
 - Bitwise matrix of roles versus privileges
 - Aggregates the access control policy of the distributed databases
- Adding yet another technology to the landscape...

Resources

- **PERMIS Home page:**
 - <http://sec.isi.salford.ac.uk/permis>
- **OASIS specifications:**
 - <http://www.oasis-open.org/specs>
- **Shibboleth**
 - <http://shibboleth.internet2.edu/>
- **NeSC support pages (“Grid Security” section):**
 - <http://labserv.nesc.gla.ac.uk/projects/etf>
 - <http://labserv.nesc.gla.ac.uk/projects/etf/gt4howto/permis.html>
- **E-mail:**
 - a.stell@nesc.gla.ac.uk

