University of Kent

Integrating VOMS and PERMIS for Superior Secure Grid Management (VPman)

Deliverable 1.2
Use cases

Version <1.0>

Revision History

Date	**Version**	**Description**	**Author**
28/May/2007	0.1	Initial draft	Bassem Nasser
8 June 2007	0.2	Second draft with comments from Rich	Richard Sinnott
11 June 2007	0.3	Updates	Bassem Nasser
21 June 2007	0.4	More Updates	Richard Sinnott
16 July 2007	0.5	More Updates	Bassem Nasser
1 August 2007	0.6	More Updates	Bassem Nasser
9 September	1.0	Final version	Bassem Nasser

Table of Content

1. Introduction

Both VOMS [1] and PERMIS [2] provide authorisation infrastructures for Grids and are currently predominantly used by different groups of Grid users. Each has its strengths and weaknesses. The VPman project plans to combine these technologies to show how improved Grid based security can be achieved drawing on the strengths of both VOMS and PERMIS. Specifically we wish to integrate VOMS attribute assignment function with the PERMIS authorisation decision function. (See deliverable D1.1 for details of the VOMS and PERMIS technologies).

A key component of this work is in ensuring that the integrated VOMS and PERMIS technologies address real needs of the Grid and e-Research communities. It is the case that several flavours of Grid middleware exist today. In the UK the most prominent of these are:

- Globus toolkit with the latest release at version 4 (GT4);
- Open Middleware Infrastructure Institute (OMII-UK);
- The pre Web Services versions of Globus as deployed across the NGS (GT2.4.3);

A key requirement is to support authorisation of services built upon these technologies. It is also the case that authorisation itself can take many different forms dependent upon the requirements of the application domains themselves. The National e-Science Centre at the University of Glasgow is involved in a rich range of projects with advanced security at their heart including the EPSRC funded pilot project *Meeting the Design Challenges of nanoCMOS Electronics* and the MRC funded pilot project *Virtual Organisations for Trials and Epidemiological Studies*. These projects will provide the primary source of requirements for the exploitation of the combined VOMS and PERMIS technologies.

The rest of this document is structured as follows. Section 2 provides and overview of the VOTES and nanoCMOS projects and section 3 describes the various interactions that will need to be supported to best utilise the combined VOMS and PERMIS software.

2. Introduction to Case Studies

2.1 Introduction to VOTES

The MRC funded Virtual Organisations for Trials and Epidemiological Studies (VOTES) project (www.nesc.ac.uk/hub/projects/votes) is a 3 year project looking at building a Grid framework for clinical trials and observational studies. The project began in October 2005 and involves the National e-Science Centre at the University of Glasgow and partners at Oxford, Nottingham, Leicester and Imperial College. Clinical trials and clinical systems more generally place many demands upon security infrastructures to support the various activities that are involved. In particular, the typical processes involved in a clinical trial will comprise recruitment, collection of data specific to the trial and overall management of the trial itself, e.g. to ensure that it is undertaken according to ethical concerns.

Fine grained security is essential in this context to ensure that the right data is made available to the right people for the right purpose. A key aspect of the work is that VOTES is not concerned with developing a single Grid infrastructure for a specific clinical trial or study, but with developing a Grid based framework through which a multitude of clinical trials can be supported.

The VOTES project has already developed initial prototypes proving proof of concept based upon existing clinical data sets and software resources used across Scotland. Versions of this framework utilize GT4, OGSA-DAI and GridSphere. The overall architecture currently supported within VOTES is outlined in Figure 1.

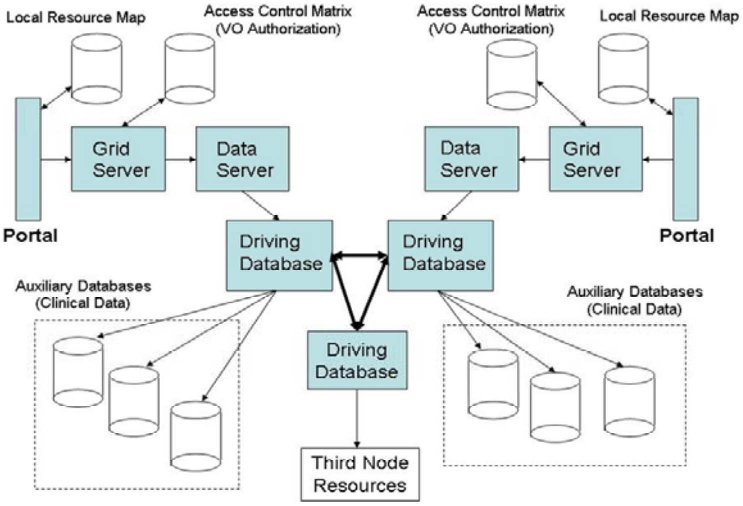

Figure 1: VOTES Data Federation Architecture

The basic interactions of the VOTES architecture are as follows:

- The user logs into the portal at a particular node - this can be directly or via Shibboleth (not shown above).
- The portal server checks the local resource files to discover the available grid servers, data servers and driving databases.
- A request containing the user's role and a specific trial is sent to the grid server.
- The grid server consults the local access matrix and returns the parameters for the resources that the user can query for that trial. These are presented to the user as a list of check-boxes, with the option to specify conditions if desired.
- The user makes their selection of parameters and submits them. These are constructed into a single query distributed across the various resources.
- The query is sent from the grid server to the data server, where it is wrapped as an OGSA-DAI service request and is passed to the driving database.
- The driving database executes the distributed query over the resources under its guard and joins the various distributed results into one single result.
- This result is sent back to the data server and then to the grid server, where it is transformed into readable HTML and finally presented to the user through the portal again.

The access matrix mentioned here is implemented through a set of relational tables within a PostgreSQL database. This delimits the roles, parameters and access rights associated with each role. The database is queried from an SQL statement, which returns the parameters relevant to that role, presented to the user in a user-friendly graphical interface. Administrative user interfaces are available which allow privileged end user, e.g. investigators in a clinical trial, to define roles and associated views of federated clinical data particular to that trial. We expect to replace this administrative interface and the access control matrix within this VPman project with VOMS specific attributes and PERMIS based authorization policies.

The details of one of the nodes of this architecture (for NeSC Glasgow) provided in Figure 2.

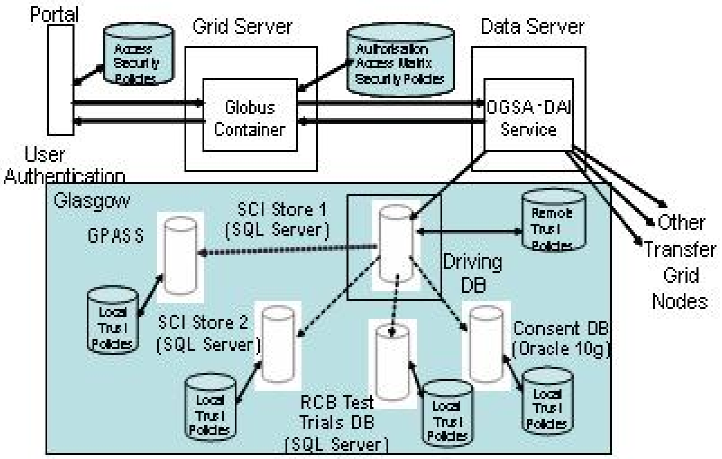

Figure 2: Architectural diagram of a data grid constructed for the VOTES Glasgow node

The technology used to implement this architecture is as follows:

- GridSphere to implement the portal that provides user-friendly access to the system.
- Globus Toolkit v4 to implement the grid service that links the UI layer to the data layer, whilst providing an intermediary layer, where the security principles can be implemented.
- OGSA-DAI is used to filter the grid service SQL query into an XML format more easily used for rendering in the portal.
- The data store used in the Glasgow site is PostgreSQL. However, this will differ across the various sites involved, depending on their local installations.

The various data resources accessible within the Glasgow VOTES node include:

- **Scottish Care Information store (SCIstore) -** is a batch storage system which allows hospitals to add a variety of information to be shared across the community, e.g. pathology, radiology, biochemistry lab results are just some of the data that are supported by SCI Store. Regular updates to SCI Store are provided by the commercial supplier using a web services interface. Currently there are 15 different SCI Stores across Scotland (with 3 across the Strathclyde region alone). Each of these SCI Store versions has their own data models (and schemas) based upon the regional hospital systems they are supporting. The schemas and software itself are still undergoing development.
- **General Practice Administration System for Scotland (GPASS) -** a client-side GP application that is in use by a large number of general practitioners, in over 890 practices in Scotland. It provides a portable, 'on-the-spot' electronic interface for GPs to input patient data, to be synchronised to the centralised SCI-Store repository at such times as a connection can be made. It also contains a local data store that houses and inter-relates information on drugs and treatments, thus providing a rudimentary ability to advise on a particular treatment, depending on patient symptoms and history.
- **Scottish Morbidity Records (SMR)** – which include good quality (linked) records relating to a variety of patient information records, compiled primarily from hospital visit records across Scotland often over an extended period. These include births, deaths, cancer registrations and other acute illnesses.
- **Other clinical trials resources** – including previously conducted clinical trials; drug dictionaries etc.
- **Consent databases** – crucial to the conduct of clinical trials is agreement (consent) by the patient to be involved in the trial itself.

This infrastructure has been applied across a variety of clinical trials in the areas of neurological studies and brain trauma; congenital anomalies; primary care recruitment and secure access to secondary care data resources, with more recent work focusing on studies of prostate cancer and type-2 diabetes.

The key scenario that we expect to use as the basis for the investigation within the VPman project is in the area of type-2 diabetes. This will require access to and usage of resources at Glasgow and at collaborating VOTES partner sites including Nottingham University and Imperial College. This work also leverages resources and software from the major UK NHS infrastructure effort: Connecting for Health.

To support this we will establish a VOMS based virtual organization (*VOMSdiabetes*) with roles associated with the various participants involved in the study (*VOMSdiabates-investigator*, *VOMSdiabetes-nurse* etc). These roles will be able to access a range of data sets associated with local site policies and VO-wide agreements. These roles will be used to provide secure access to and usage of the federated data sets through PERMIS based authorization decisions. In the first instance we expect that this will be achieved through the definition of fixed stored procedures on the driving database, i.e. the queries are themselves fixed. However as with previous trials, we support the parameterization of these queries – thus a privileged user may enter specific ranges or strings which can constrain the query. These stored procedures will be used as the basis for the definition and subsequent enforcement of the authorization policies.

We note that these stored procedures will result in federated subqueries being generated which will retrieve data from a variety of data resources including SCIstore (diabetes lab results), GPASS (diabetes patient records), SMR (diabetes historical data) and data repositories in Imperial and Nottingham.

This VOTES scenario will thus explore how GT4 services can be accessed and used through VOMS and PERMIS technologies.

2.2 Introduction to nanoCMOS

The EPSRC pilot project *Meeting the Design Challenges of nanoCMOS Electronics* (nanoCMOS) started in October 2006 (www.nanocmos.ac.uk). This project is lead by the University of Glasgow with the e-Science component lead by NeSC Glasgow. This project will provide a rich vein of scenarios which will be explored in this project.

The electronics domain demands infrastructures that support protection of intellectual property, be it for designs of transistors, data sets, simulation codes. The quantum level effects of devices of ever decreasing dimensions are becoming ever more important and atomistic simulation of devices is necessary. The nanoCMOS project plans to develop an infrastructure through which device level designs and simulations can be linked through to higher level circuit and system simulations, to predict the overall behaviour of systems.

This will require both large scale HPC usage including access to and usage of the NGS, and will generate many terabytes of data. The project is using OMII-UK for Grid enabling simulations, using the OGSA-DAI components of OMII-UK for data access and integration, and the OMII-UK myGrid components to define and enactment workflows.

Diagrammatically, the overall sets of services and data are depicted in Figure 1. We expand on these services and data sets to the extent that they impact on the design and development of the Grid infrastructure. We note that for each of these services numerous instances using different technology bases can exist including both commercial offerings and academic software.

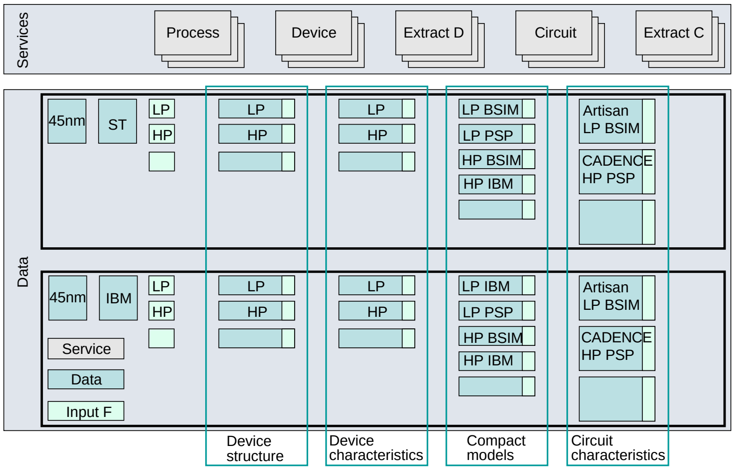

Figure 3: Conceptualisation of the nanoCMOS Services, Data and Design Processes

It is the case that each of services (which we outline below) and data sets has strict intellectual property rights associated with them. These data sets themselves can be associated with particular technology nodes, e.g. 45nm scale devices from IBM; targeted towards particular types of device, e.g. low-power (LP) or high-power (HP); aimed at BSIM or PSP models.

The services and data sets themselves support the complete spectrum of processes associated with electronics design: process simulation, device simulation, compact model extraction and circuit simulation.

nanoCMOS::Process simulation

The process simulation is concerned with the physical steps necessary to turn a piece of silicon into a working device. This might include for example dopant implantation information; oxide growth; etching, deposition of metals etc. It is typically the case that this information is supplied by a commercial manufacturer such as IBM or Synopsis. This information is provided typically as a .tif file. In future it may well be possible to generate these files directly through simulation. Currently the models explored in nanoCMOS assume this file is provided as an input.

nanoCMOS::Device simulation

A device simulation itself typically involves the solution of sets of coupled equations describing the distribution and flow of electrons in a given device structure. These might be based upon Drift Diffusion approaches solving Poisson’s equation, or based on other approaches such as Monte Carlo based ab initio simulations.

The inputs to a device simulation will be information from the process simulation, i.e. the provided (or generated) .tif file, the dopant profile extracted for this device and further input file including information such as the simulation mesh used as the basis for the device simulation, and the oxide or nitride coating used with this particular device. Multiple (ensembles!) of device simulations need to be run to characterise the behaviour of a particular device. Each of these will have slightly different dopant profiles reflecting the variability inherent due to atomic scaling.

The output of a given device simulation is a current/voltage (I/V) curve describing the characteristics of that device with that particular dopant profile. The device simulation process as a whole generates many such I/V curves characterising the behaviour of that device with different dopant concentrations and distributions.

The device simulations themselves can be based on commercial tools such as Taurus^{TM} or based on academic developed solutions. In the commercial case, license management is a crucial aspect to address for the Grid infrastructure.

The device simulations themselves can be extremely computationally intensive and generate extremely large data sets. As a result, a key requirement is to be able to access and use large scale computational resources such as the National Grid Service (www.ngs.ac.uk) and ScotGrid (www.scotgrid.ac.uk).

The simulation software itself is primarily based upon Fortran 90 code.

nanoCMOS::Compact model extraction

Having generated the set of I/V curves for a particular device, it is necessary to abstract this information to a higher level so that multi-device circuit/system simulations can be performed. Compact models are semi-empirical analytical descriptions of the response of a device.

This process of generating a compact model is achieved through identification of an extraction strategy (identifying the subset of device model parameters which most influence the curves) and exploitation of commercial tools such as Aurora^{TM}. It is the case that this phase will typically require domain knowledge and expertise in identifying the particular parameters that most influence the generated I/V curves.

This phase will typically require access to a larger shared memory type CPU resource.

nanoCMOS::Circuit simulation

Once compact models have been generated they can be used by circuit simulators to predict the behaviour of circuits and systems built from multiple combinations of these compact models. Typical examples of the kinds of behaviour analysed at the circuit/system level with these compact models are to identify how the set of connected components respond to a stepped input voltage or to explore particular tolerances of the integrated circuit.

Feedback at this stage can require modifications to the generated compact models which in turn may require device simulations to be redone. Linkage of device simulations, compact models and circuit/system simulations are an essential component of the nanoCMOS project in understanding how atomistic variation of devices impact upon system level design and simulation.

Both commercial and non-commercial applications are used for circuit simulation. One of the most widely used circuit simulators today is the Simulation Program with Integrated Circuit Emphasis (SPICE) simulator which has both open source and licensed versions.

nanoCMOS Grid Developments

The nanoCMOS proposal recognised four key aspects that the Grid infrastructure must address. These include: security; data management; workflows and resource management. At the time of writing work has focused primarily upon security and development of core simulation services supporting the processes identified in Figure 3.

It is the case that each of the services identified requires simulation tasks to be completed. In the first year of the project we have decided to demonstrate at least one service for each of the simulation tasks: device simulation, compact model extraction and circuit simulation.

For the development of the Grid infrastructure the project has aligned itself with the OMII-UK technologies (www.omii.ac.uk). The data management and workflow aspects of the OMII-UK project are still at an early phase and work has focused initially upon the development of a family of OMII-UK services which support the device modelling and compact model generation phases of electronics design. These services have been developed to exploit the OMII-UK GridSAM job submission system.

The initial nanoCMOS services themselves that have been prototyped thus far in nanoCMOS have included the first version of a device modelling simulator (called Geronimo) and a Grid enabled version of a commercial application Aurora used for creation of compact models. The development of a SPCE simulator is also underway.

In the context of VPman the nanoCMOS project will require the establishment of a VOMS based virtual organization (nanoCMOS) with roles defined pertinent to the needs of the electronics community. These might include for example *nanoCMOSdeviceSimulator* and *nanoCMOScircuitSimulator* roles. We will show how VOMS attributes specific to the teams involved in nanoCMOS design can be defined and subsequently used by OMII services in combination with PERMIS to enforce policy decisions. This will include policies on access to OMII based HPC services accessing the NGS compute nodes; policies on electronic data sets hosted by NGS data nodes, and policies on resource usage across NGS and other HPC nodes across partner sites.

To understand the way in which VOMS and PERMIS will be exploited by these two projects, we consider the following sets of interactions. In the first instance we outline the scenarios without use of VOMS and then add with VOMS.

3. Interaction Scenarios for Case Studies

3.1 VOTES Diabetes Trial with basic GT4-Authorisation without VOMS

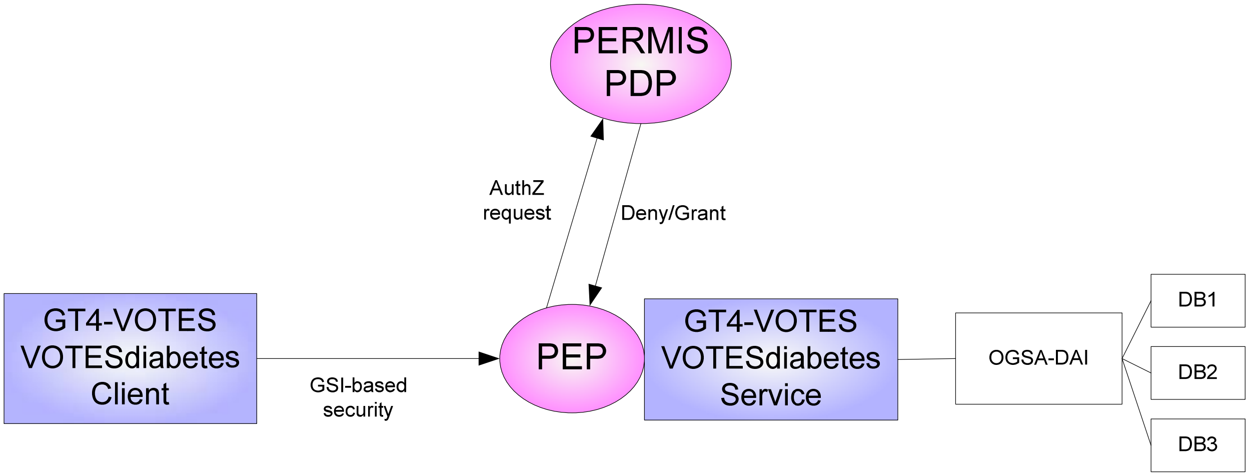

Figure 4: VOTES Diabetes use case without VOMS

The interactions to support security of the underlying clinical data resources are as follows (assuming that the GT4 based services and data resources have been pre-deployed along with appropriate security policies):

- A user runs “grid-proxy-init” to generate a proxy certificate and tries to invoke the VOTESdiabetes stored procedure
- Through information in the service deployment descriptor .wsdd, the PEP can detect if authorisation decision is needed.
- The PEP picks up DN from the certificate and calls the PERMIS PDP for decision via its XACML interface.
- The PERMIS CVS (Credential Verification Service) pulls the DN attribute certificates from a repository and validates them.
- The PERMIS PDP according to the policy, user request and user attributes, decides if the request is to be granted or denied.
- If the user is authorised they are able to invoke the stored procedure (via GT4 and OGSA-DAI) and the result set is returned to the end user

3.2 VOTES Basic GT4-AuthZ with VOMS Push model and no Portal

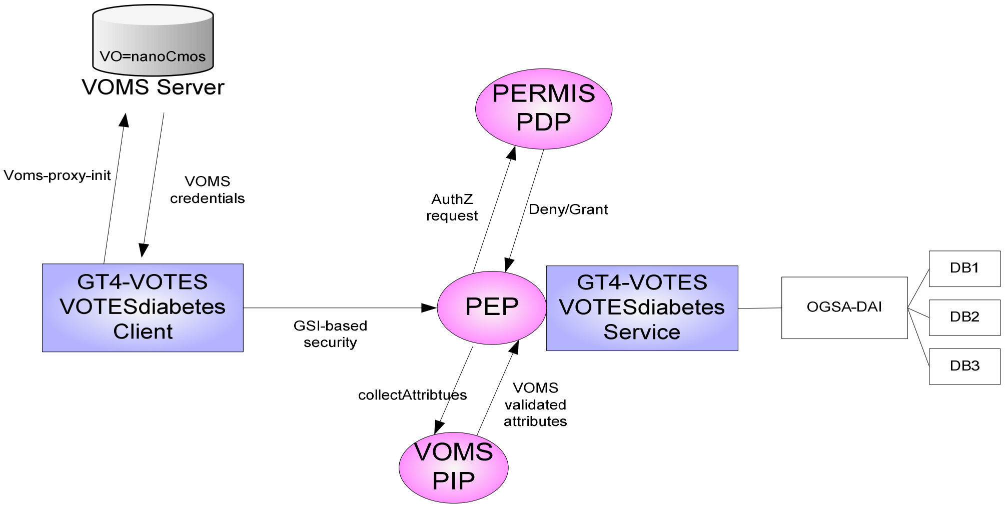

Figure 5: VOTES Diabetes use case with VOMS

The scenario works as following:

- A VOTES diabetes service is deployed on a GT4 infrastructure
- A user runs “voms-proxy-init” to generate a proxy certificate including VOMS credentials and tries to invoke the protected stored procedure
- The PEP passes the user information (including proxy certificate) to the VOMS PIP
- VOMS PIP validates the credentials and passes back the VOMS Fully Qualified Attribute Name (FQAN) within the subject attributes.
- The PEP calls the PERMIS PDP pushing the request information and credentials
- The PERMIS PDP according to the policy decides if this user with certain attributes is authorized to access the service.
- If successful the stored procedure is invoked, the federated query run and returned results joined and returned to the end user

3.3 nanoCMOS Basic OMII-AuthZ with no VOMS

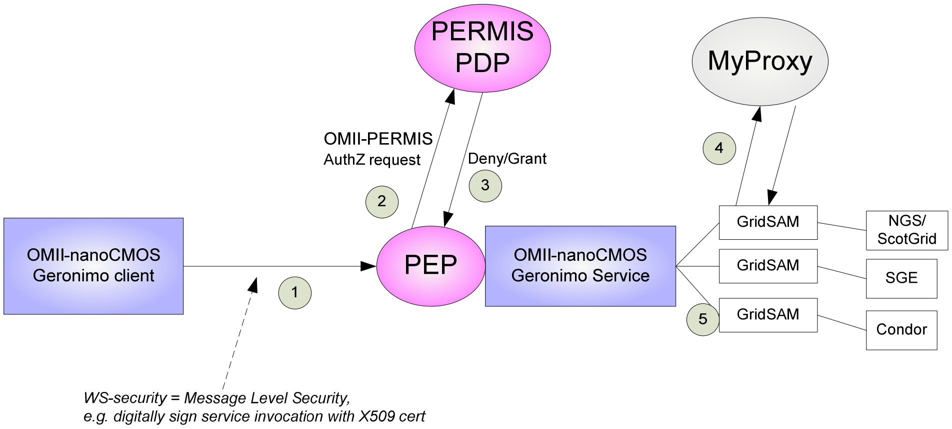
PERMIS PDP
MyProxy
OMII-PERMIS AuthZ request
2
Deny/Grant
3
4
OMII-nanoCMOS Geronimo client
1
PEP
OMII-nanoCMOS Geronimo Service
GridSAM
GridSAM
GridSAM
5
NGS/ ScotGrid
SGE
Condor
WS-security = Message Level Security,
e.g. digitally sign service invocation with X509 cert

Figure 6: nanoCMOS use case without VOMS

In this scenario:

- The user signs client invocation with his X509 certificate to invoke for example the Geronimo service
- Through information in the service deployment descriptor .wsdd, the PEP can detect if an authorisation decision is needed.
- The PEP extracts DN of user from the SOAP message (or some other uniquely identifying information!!!) and passes this to the PDP with details of request.
- The PERMIS PDP makes decision and if the user is authorised the job is submitted to the GridSAM instance which submits the job on behalf of the user to the appropriate resources
- Note that the policy here might indicate obligations as “where a users job is allowed to run” and not only whether they can see/invoke the service
- Note that GridSAM may also require the users proxy certificate to be created on the server side (potentially by MyProxy).
- The results are then returned to the client

3.4 nanoCMOS Basic OMII-AuthZ with VOMS and Client Push for nanoCMOS Project

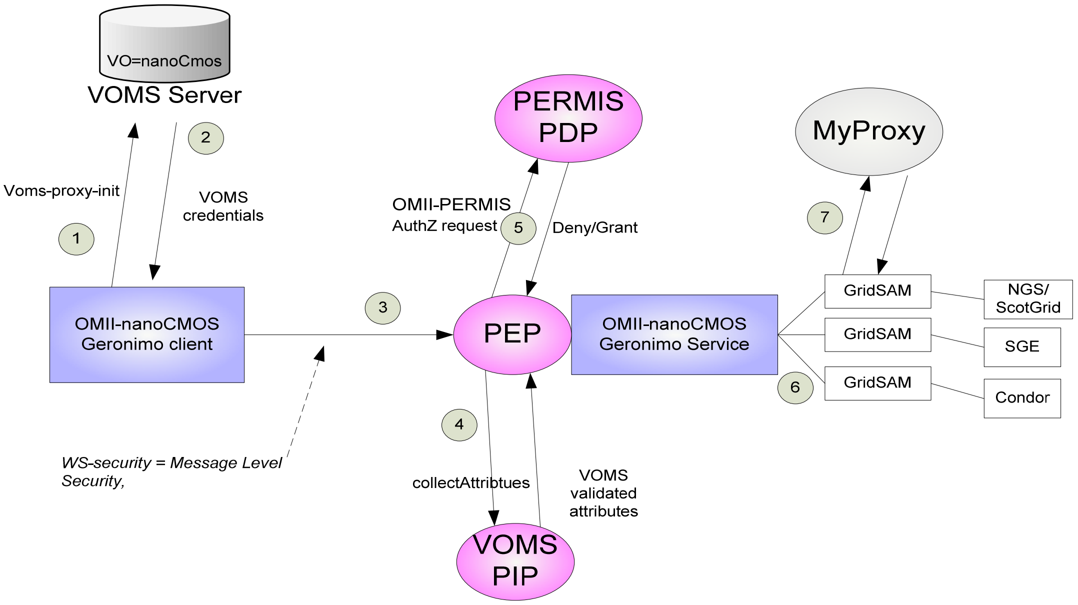

Figure 7: nanoCMOS use case with VOMS in push mode

In this scenario VOMS is included:

- A user runs “voms-proxy-init” to generate a proxy certificate including VOMS credentials VO=nanoCMOS, Role=NanoCMOS_deviceModeller
- An X509 proxy cert is created with VOMS AC’s appended
- The user attempts to submit a job to the Geronimo service
- Through information in the service deployment descriptor .wsdd, the PEP can detect if authorisation decision is needed.
- The VOMS PIP validates the credentials and passes back the VOMS Fully Qualified Attribute Name (FQAN) within the subject attributes.
- The PERMIS PDP makes the authorisation decision
- If ok, the details of the job are submitted using the supplied X509 cert via GridSAM to appropriate Grid resources and return results to client
- If job is allowed on NGS/ScotGrid,, it will need to obtain suitable authorisation credentials. This may involve either activating a certificate from MyProxy, or passing on VOMS ACs to the NGS so that it can utilise them for authorising the access, via its own PERMIS PDP and Policy.
- The results are then returned to the end user

3.5 Basic OMII-AuthZ with VOMS and Service Pull for nanoCMOS Project

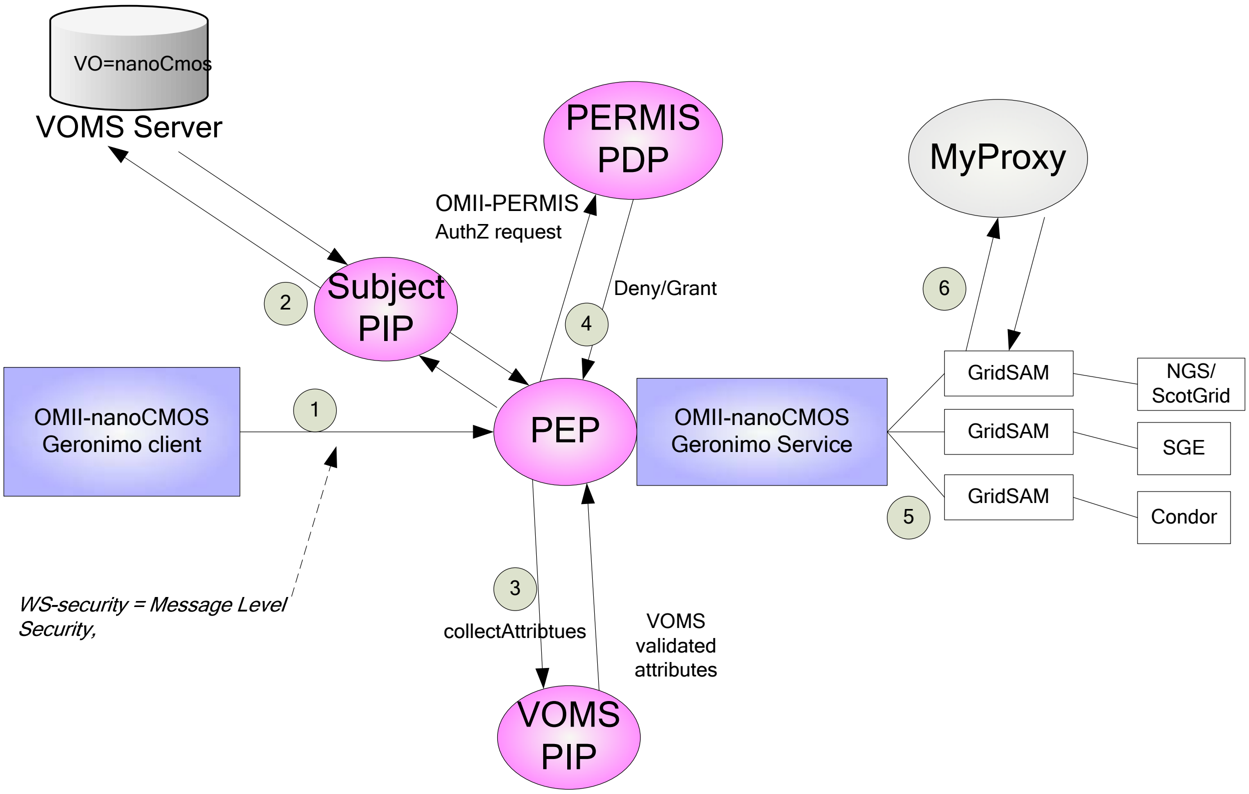
VO=nanoCmos
VOMS Server
PERMIS PDP
MyProxy
OMII-PERMIS AuthZ request
Deny/Grant
2
Subject PIP
4
6
GridSAM
NGS/ ScotGrid
OMII-nanoCMOS Geronimo client
1
PEP
OMII-nanoCMOS Geronimo Service
GridSAM
SGE
GridSAM
5
Condor
WS-security = Message Level Security,
3
collectAttribtues
VOMS validated attributes
VOMS PIP

Figure 8: nanoCMOS use case with VOMS in pull mode

This use-case consists of:

- The client attempts to invoke the PERMIS protected Geronimo service
- The PEP extracts the users DN and identifies that it needs attributes from a VOMS server (we consider this static for now and will not deal with users in multiple VOs)
- The PEP, via a Subject PIP, pulls back the relevant attributes from VOMS server and passes them to the PDP (assuming that a protocol for pulling VOMS attributes has been chosen and PERMIS has been extended to support this protocol)
- The PERMIS PDP makes the decision
- If ok, submit job using via GridSAM to appropriate Grid resources (possibly utilising suitable authorisation credentials such as a MyProxy certificate or VOMS ACs), and return the results to the client.

3.6 Basic GT4-AuthZ with VOMS in Pull Mode

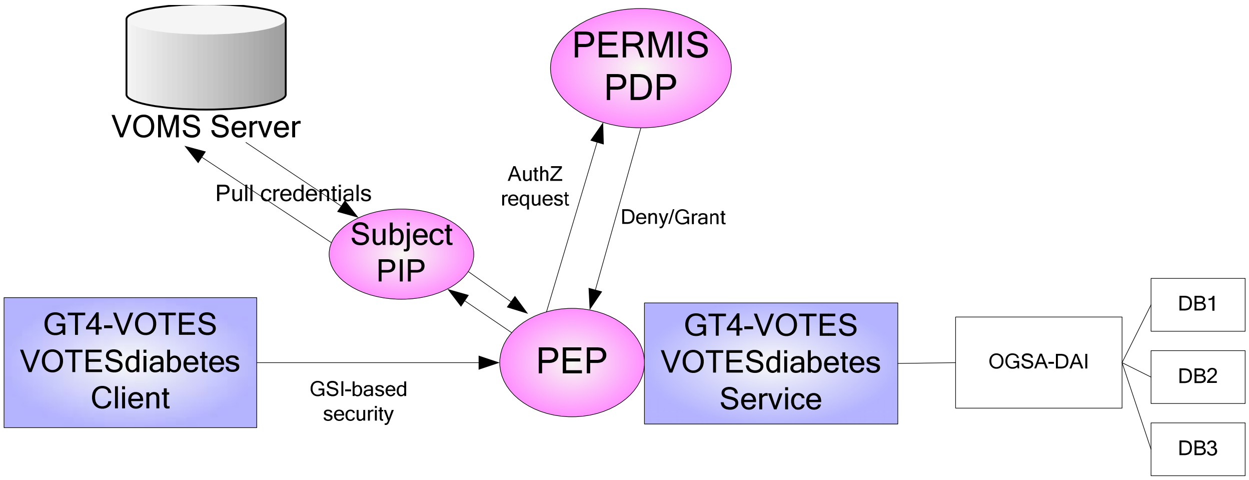

Figure 9: VOTES Diabetes with VOMS in pull mode

- A user runs “grid-proxy-init” to generate a proxy certificate and tries to invoke the VOTESdiabetes stored procedure
- Through information in the service deployment descriptor .wsdd, the PEP can detect if authorisation decision is needed.
- The PEP extracts the users DN and identifies that it needs attributes from a VOMS server
- The PEP, via a Subject PIP, pulls back the relevant attributes from VOMS server and passes them to the PDP
- The PERMIS PDP makes the decision

3.7 Shibboleth protected resources

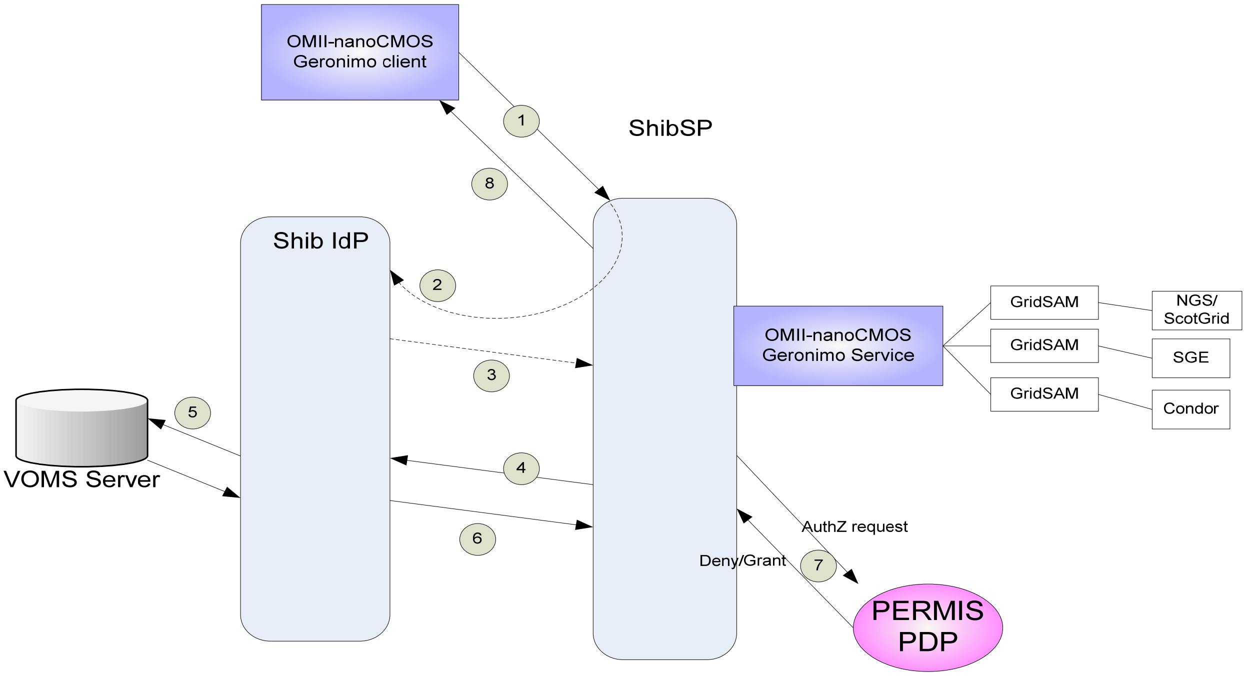

Figure 10: Shibboleth protected resources with VOMS

In this scenario shibboleth is included. A Gernonimo service is protected by Shibboleth SP. The scenario works as follows:

1- A Gernonimo client tries to invoke the service
2- The client is redirected to the IdP to be authenticated (possible via a WAYF)
3- The IdP send the authentication information to the SP
4- Upon successful authentication the SP queries the ShibIdP AA for the user's attributes
5- The IdP queries the VOMS server for user's VOMS credentials.
6- The IdP forwards the attributes to the SP
7- SP calls PERMIS for authorisation decision
8- The user is granted/denied access to the requested service

4. Conclusion

Both VOMS [1] and PERMIS [2] provide security management infrastructures for Grids but are predominantly used by different groups of Grid users. Each has its strengths and weaknesses and their combination would be a powerful solution to Grid security management.

We believe that integrating VOMS and PERMIS, more specifically the VOMS user management and attribute assignment function with the PERMIS policy based authorisation decision function; is of great benefit to the grid community users, providers and administrators. Moreover, the integration work is motivated by the use-cases presented in this document and being required by the project partners in order to enable fine-grained policy-driven authorisation via PERMIS.