DataGrid

VOMS vs EDG Security Requirements

WP06

Document identifier:	
Date:	**12/11/2002**
Work Package:	**WP06**
Partner(s):	
Lead Partner:	
Document status:	**DRAFT 0.1**

Abstract

This document lists the answers of VOMS to the Security Requirements specified in the D7.5 Deliverable.

Delivery Slip

	Name	Partner	Date	Signature
From				
Verified by				
Approved by				

Document Log

Issue	Date	Comment	Author
0-1	12/11/02	First draft.	Authorization WG

CONTENTS

1. INTRODUCTION

1.1. OBJECTIVES OF THIS DOCUMENT

In the following section we list the Security Requirements from the Deliverable D7.5 (Document Identifier: DataGrid-07-D7.5-5-0111-4-0), specifying if they are satisfied by the VOMS System and why.

The indented paragraphs are quotes from the D7.5 Deliverable.

1.2. APPLICABLE DOCUMENTS AND REFERENCE DOCUMENTS

1.2.1. Applicable documents

[A1] Security Requirements and Testbed 1 Security Implementation. [D7.5]

DataGrid documents can be found by going to the DataGrid web page at **http://eu-datagrid.web.cern.ch.**

1.3. TERMINOLOGY

VOMS certificate	A standard GSI user's proxy certificate which contains in an extension an attribute certificate signed by a VOMS server, specifying user's attributes.

1.4. GLOSSARY

ACL	Access Control List
CA	Certificate Authority
CAS	Community Authorization Service (developed by Globus)
CPS	Certificate Practice Statement
CRL	Certificate Revocation List
DN	Distinguished Name (in the X509 certificates)
EDG	European DataGrid Project
GIIS	Grid Index Information Service (Globus)
Globus	A US project developing Grid Middleware (http://www.globus.org)
GRIS	Grid Resource Information Service (Globus)

GSI	Grid Security Infrastructure, Globus's implementation of GSSAPI
GSSAPI	Grid Security Service APplication Interface see RFC2743
HTTP	HyperText Transmission Protocol
LDAP	Lightweight Directory Access Protocol
OCSP	Online Certificate Status Protocol (OCSP) enables applications to determine the (revocation) state of an identified certificate (RFC 2560).
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure X.509 working Group (http://www.ietf.org/html.charters/pkix-charter.html)
RA	(Certificate) Registration Authority
RFC	IETF Requests For Comments (http://www.ietf.org/rfc.html)
RM	Replica Manager
SE	Storage Element
SSL	Secure Socket Layer
TLS	Transport Layer Security
UID	User Identity – often called user name in conventional computing
VO	Virtual Organisation
WG	Working Group
WP	Work Package (in DataGrid)
X.509	X.509 certificate based on PKI

2. REQUIREMENTS

2.1. AUTHORIZATION REQUIREMENTS

AUZ-01 **Users may be members of any number of Virtual Organisations.**

Satisfied

AUZ-02 **Users may have any number of roles within a given Virtual Organisation.**

Satisfied.

Users may have any number of roles and belong to any number of groups.

AUZ-03 **The Virtual Organisation must be able to decide user membership policy and user authorization policy.**

Satisfied

User membership is controlled by the VO administrators. User authorization is tied to which groups and roles the user belongs (according to agreements between the VO and the local sites).

CNF-16: **A user's membership of Virtual Organisations must be confidential information.**

It should not be possible for anyone (or any process) to generate the list of virtual organisations to which a person belongs, but it must always be possible to verify that a person is a member of an organisation to which they claim membership, this is necessary for authorization to work. It should also not be possible for anyone other than the managers of a Virtual Organisation to produce a list of members of that Virtual Organisation. This implies that a user will need to give authorization to check that he is a member of a VO, and has certain roles within that VO, in order to gain permission to carry out the relevant actions.

One reason for this is that membership of 2 VOs may be in conflict, and it should not be possible for 1 VO to determine whether a person is a member of another VO.

The organisation to which the user belongs is in the user's VOMS certificate.

For compatibility reasons (i.e. *grid-mapfile* generation) it is possible to extract from the VOMS Servers the list of users. The access to this information can be restricted by checks on the certificates which the queerer must present.

AUZ-05; **The owner of a resource or data should be able to allow users authorization to carry out an action based on any of the following:**

1. **by public access**
2. **by only having acceptable authentication**
3. **by membership of a VO**
4. **by role(s) within a VO**
5. **by membership of a combination of VOs and roles**
6. **by allowing selected certificates**

7. **by personal User ID on the system**
8. **individual certificates may be banned.**

VOMS certificates contains — among others — user's identity (the DN of his certificate), VO, group and role membership information. This allows authorization according to the methods 2, 3, 4, 5, 6 and 8.

AUZ-06: **The authorization method must allow any combination of the above authorization requirements, including any combination of VOs and roles**

VOMS certificates may contain user membership info (groups and roles) for an arbitrary number of VO's.

AUZ-07: **There should be no restrictions on the degree/level of granularity of authorization. In particular, no hard-coded limits to how the granularity is set should exist.**

This should include e.g. allowing authorization to a hierarchy of directories, individual directories, or individual files

A producer (of information) must be able to restrict read access to information to specific authorised users. Access must be controllable at a level of granularity specified by the intersection of a set of rows as expressed by SQL WHERE clauses and a set of columns.

N/A

AUZ-08: **There should be the possibility for certain roles to have further authentication, such as an additional password.**

N/A

AUZ-09: **It must be possible to assign priorities to jobs, within the range and the resources the user is allowed to access by the VO. This must include immediate processing of jobs, interrupting or suspending running jobs if needed.**

N/A

AUZ-10: **It must not be possible to *successfully* carry out an action or *successfully* submit a job where the authorization is not valid.**

N/A

AUZ-11: **It should be possible to determine the list of resources to which a user has access and what actions they are allowed to carry out in the VO(s) and role(s) set for the current session.**

N/A

AUZ-12: **It should be possible to determine if a certain user in the role(s) and VO membership(s) set for the current session has access to a certain resource and what actions they may carry out on that resource.**

N/A

AUZ-13: **At the time a job is submitted, the resource broker should only submit to resources the user is authorized to carry out the requested actions in the role(s) and VO membership(s) set for the current session.**

N/A

AUZ-14: **The authorization method must be application independent.**

N/A

AUZ-15: **The authorization decision making process must be the same/consistent within a VO**

For example, if there is more than 1 authorization server the authorization decision must not be different according to which server is used.

If a user has authenticated themselves by use of a certificate, and claims membership and roles within a VO, then the authorization decision must not depend on where the user initially logged on.

N/A

AUZ-16: **There should be local authorization servers at every site so that the local services can continue their operation even if a site is disconnected.**

N/A

AUZ-17 **It must be possible to disable a user's authorization in the following ways:**
1. it must be possible to remove a user from a VO
2. it must be possible to remove a given role or a number of roles from a given user.

Satisfied.

AUZ-18: **It must be possible to disable a user's authorization within 10 minutes.**

Not satisfied.

The user can be removed from the VO (see **AUZ-17**), but there is no method to invalidate VOMS certificates already issued and still valid. However, the validity of a VOMS certificate can be made as short as desired (it is a parameter independendent from the user's proxy lifetime, and decided by the VO manager).

AUZ-19: **When a user's authorization to carry out certain actions has been disabled, it must not be possible to carry out any action that is no longer authorized.**

N/A

AUZ-20: **When a user is in a given role within a VO, the VO must be able to specify security requirements on any resources where actions are carried out.**

For example, if a user is processing confidential data, that data should only be placed on resources having adequate security to handle that data. The VO should be able to specify what level of security resources should have when the user is in a confidential role.

In some cases, the VO may specify that the job may only be executed on its own resources.

If the confidentiality requirements are *always* associated with the role, this is a matter dealt with by agreements between VO and local sites.

As an alternative, it could be used the "capability" attribute, to specify the requirements for special processing.

However, at least at the moment, there is no way to force the specification of a particular attribute. In other terms, it is the user who *asks* to be certified as owner of certain attributes, and he can select the ones he wants.

The VO – via ACL's – can instead specify special processing requirements on the local resources.

AUZ-21: After the user has authenticated himself, the user must be able to select and de-select VOs and roles. Permissions must be granted automatically and security requirements must be adhered to automatically.

Satisfied

The user can generate as many proxies he needs, each with the attributes he needs from the VO's he chooses.

AUZ-22: It should be possible to determine the list of resources which have sufficient authorization to carry out actions when the user is in the VO(s) and role(s) set for the current session.

N/A

AUZ-23: It should be possible to determine if a certain resource has sufficient authorization to carry out actions when the user in the role(s) and VO membership(s) set for the current session

N/A

AUZ-24: When a user carries out processing in a secure role, the job should only be submitted to resources which are authorized to carry out that job, and store data on storage elements with suitable authorization.

N/A

AUZ-25 : It should be possible to base authorization on any one of the following, in addition to the authorization requirements in AUZ-05:

1. **Role (A special case of a role may be a group, a role may also be mapped onto e.g. a unix group id)**
2. **File name**
3. **Storage element name**
4. **Operation (including metadata and file operations)**
5. **Resource usage limits. (E.g. quota)**
6. **Directory**

It should be possible to put a number of specific users in a group, and allow these joint access to certain files. This group may be e.g. a subgroup of a Virtual Organisation, or a list of DN's which require access to certain files, or a list of UID's on the system.

Satisfied by appropriate ACL's on the local resources

AUZ-26: The authorization requirements on file access should hold regardless of replication.

N/A

AUZ-27: The Authorization mechanism must preserve the identity of the user, i.e. the DN or distinguished name of the user.

Satisfied.

The VOMS certificate contains the DN of the user's certificate.

AUZ-28: It should be possible to assign a user to set the authorization on file access.

This is rather like the conventional file owner, or can be considered to be an administrator of file access rights. This user can allow or deny others permissions on the file. If a user carrying out research generates a file, that user may make that file available to a specific user they are collaborating with. If the file is part of a database, which is owned by a Virtual Organisation, then it should be possible to only allow VO members in the role of database managers permission to delete those files.

N/A

AUZ-30 : If files are replicated, authorization for access to this replicated data must not depend on one other single site being available.

N/A

AUZ-31: An unauthorized user must not be able to delete or alter a file for which he has no access right.

N/A

AUZ-32: It must be possible to confirm that a user has the VO membership(s) and Role(s) they are claiming at the time they request an action.

Not satisfied.

At the moment there isn't an online "revocation" checking protocol (a la OCSP)..

AUZ-33: The VO should be able to specify a list of either which specific resources, or which specific VO's resources are acceptable when a user is in a particular role.

See the answer to **AUZ-20**.

AUZ-34: All applications must be able to set authorization to carry out different actions.

N/A

2.2. CONFIDENTIALITY REQUIREMENTS

The requirements not listed aren't relevant for VOMS.

CNF-15: The list of users and their E-mail addresses must be only available to a small number of authorized people.

See the anwer to **CNF-16** in the *Autorization Requirements* Section.

CNF-16: A user's membership of Virtual Organisations must be confidential information.

This requirements is listed in the *Autorization Requirements* Section.

2.3. USABILITY REQUIREMENTS

The requirements not listed aren't relevant for VOMS.

USR-08: Tools should be available to allow the user to select and de-select which VOs and roles they wish to enable for the current session. This should include the ability for each user to select a default VO and role which should automatically be set up at the beginning of each session.

We envisage something like the following:

1. the user authenticates himself.
2. the user selects which VO(s) he wishes to enable for the current session.
3. default role(s) are then selected
4. the user selects any additional role(s) they wish to enable.
5. the system automatically grants relevant permissions.
6. the system automatically adheres to the security authorization on resources required by the roles
7. the user should also be able to disable roles and VOs.

Points 1 to 4 are accomplished by the **voms-proxy-init** command, which produces a VOMS certificate.

For points 5 and 6 see the answer to **AUZ-20**.

Points 7 is accomplished by destroying the VOMS cerificate.