How the authorization information is used in matchmaking

The following expression is evaluated at matchmaking time in order to check whether the owner of a job has access rights to a given CE.

```
AuthorizationCheck = (
   member(other.CertificateSubject, GlueCEAccessControlBaseRule) ||
   member(strcat("VO:",other.VirtualOrganisation), GlueCEAccessControlBaseRule) ||
   FQANmember(strcat("VOMS:",other.VOMS_FQAN), GlueCEAccessControlBaseRule)
) &&  ! FQANmember(strcat("DENY:",other.VOMS_FQAN), GlueCEAccessControlBaseRule);
```

We check if either the certificate subject or the virtual organization the user belongs to is member of the *GlueCEAccessControlBaseRule* (ACBR henceforth in text) of the CE.

The third expression in logical OR condition has been added in order to support generic attributes specification in the ACBR and tests for ownership of the primary FQAN specified in the user-proxy. The *VOMS_FQAN* attribute in the JDL is assigned with such a value.

The classad built-in member function, while testing for ownership in the ACBR list, uses a lexical match (classic string compare). The *FQANmember* function as the list mernership built-in fuction *member(V,L)* takes two arguments: the FQAN and the list of ACBR. The *FQANmember* returns `true` if and only if the FQAN is a member of the ACBR list and uses an ad-hoc comparator while testing for ownership. A detailed description of the FQANmember function in classad syntax is given as follows:

```
BOOLEAN VALUE FQANmember(fqan, acl)
```

where:

- `fqan` is either a *LITERAL NODE* or an *ATTRREF NODE* which should evaluate to string representing the Fully Qualified Attribute Name (FQAN) we are performing the access control list membership for;
- `acl` is an *EXPR LIST NODE* or an *ATTRREF NODE* which should evaluate to a list of string representing the Access Control rules we are matching the first argument (fqan) against.

The MM *receives* the authorization information i.e. ACBR from the classad representation of a CE, which is generated starting from the information the BDII publishes for that CE. The WMS system queries the BDII in order to gather a representation of Grid resources to be cached in the Information ISM. Information purchasers acquire information about both CEs and SEs. With respect to the Computing Element information, the following objectclasses are involved: *GlueCE*, *GlueCESEBind*, *GlueCluster*, *GlueSubCluster*. After the introduction of the VOViews the ISM purchaser has been modified in order to also query *GlueVOView* objectclass and process information about VOView according to the GLUE Schema 1.2 specification. For each defined VOView, a ClassAd representation of the CE is generated, merged with the VOView attributes and finally inserted in the ISM.

In other words, for each VOView defined for a CE, the system inserts a ClassAd that is constructed by

- taking the CE information
- replacing the CE info with VOView info if published (i.e. EstimatedResponse-Time)
- replacing the CE ACBR with the VOView ACBR

It should be pointed out that the orginal space of authorization rules should be preserved. After processing all the VOView for a given CE, we check whether there are ACBRs in the generic CE block which were not present in any VOView block (not mapped to any Views), or not. Accordingly, a final CE ad which comprises the CE information along with the list of the *orphan* CE ACBRs is generated and inserted in the ISM.

As an example let's consider the following scenario where a computing element providing access to tree different VOs has only two of these VOs bound to voviews:

```
dn: GlueCEUniqueID=wn-04-01-03-a.cr.cnaf.infn.it:2119/jobmanager-lcglsf-cms,
mds-vo-name=local,o=grid

[...]
GlueCEAccessControlBaseRule: VO:cms
GlueCEAccessControlBaseRule: VO:atlas
GlueCEAccessControlBaseRule: VO:gilda
[...]

dn: GlueVOViewLocalId=cms-view,GlueCEUniqueID=wn-04-01-03-a.cr.cnaf.infn.it:2119/
jobmanager-lcglsf-cms,mds-vo-name=local,o=grid

[...]
GlueCEAccessControlBaseRule: VO:cms
[...]

dn: GlueVOViewLocalId=atlas,GlueCEUniqueID=wn-04-01-03-a.cr.cnaf.infn.it:2119/
jobmanager-lcglsf-atlas,mds-vo-name=local,o=grid

[...]
GlueCEAccessControlBaseRule: VO:atlas
[...]
```

Clearly, the space of authorization rules the computing element provides, is not entirely covered by the rules the Views supplies with. In such a case, in order to authorize users matching the third access control base rule, also the initial computing element classad specification should be inserted in the ISM and the GlueCEAccessControlBaseRule accordingly modified. -- FrancescoGiacomini - 09 Oct 2007

This topic: EgeeJra1It > AuthZinMM
Topic revision: r4 - 2007-10-24 - SalvatoreMonforte