

# The SAM-Grid security model for JIM V1

Gabriele Garzoglio for the JIM Team.  
5/28/03

## Abstract

This document presents a discussion of the security model of the SAM-Grid our proposed to implementation for JIM V1.

Abstract .....	1
Overview of the SAM-Grid architecture.....	1
The JIM V1 security model.....	2
Client Site.....	3
Submission Site.....	3
Execution Site .....	4
Resource Selector.....	4
Trusted CA and maintenance of the identity lists .....	4
Issues related to the FNAL security policy .....	4
Proposed solution to maintain the identity lists .....	5

## Overview of the SAM-Grid architecture

The SAM-Grid is a software suite that addresses the globally distributed computing needs of the Run II experiments at Fermilab. The Job and Information Management (JIM) components complement the Data Handling system of the experiments (SAM), providing the user with transparent remote job submission, data processing and status monitoring.

The logical entities of the SAM-Grid consist of

1. multiple Execution Sites
2. a central Resource Selector<sup>1</sup>
3. multiple Job Submission Sites
4. multiple Clients (User Interface) to the Job Submission Sites.

Servers at the Job Submission Sites and at the Execution Sites register with the Resource Selector. Users describe and submit jobs to the Submission Sites via a User Interface, ultimately installed on a laptop. The Submission Sites maintain a queue of jobs that are periodically matched with the available resources. Matches are currently ranked by the Resource Selector according to the number of files of interest to the job that are already present at the Execution Site. Submission Sites are then responsible to reliably dispatch the job to the Execution Site.

Typical resources at the execution site consist of

1. A Local Resource Management system
2. A SAM Station
3. An Information Manager

---

<sup>1</sup> The Resource Selector can be in principle distributed; for the deployment of JIM V1 it will be central, though.

The Local Resource Management system generally has experiment specific interfaces<sup>2</sup> and is based on a Batch System; it is responsible to receive and process jobs from the Submission Site. The SAM Station is a collection of resources managed by a set of services to satisfy Data Handling requests from individual jobs or other entities, like the Information System or the Resource Selector. It generally manages a pool of disk caches and may be interfaced to a local Mass Storage System. SAM Stations rely on a set of supporting services, some of which are distributed some are central. The Information Manager provides service configuration support and monitoring of status information. Each Site advertises resource availability to the Resource Selector. Figure 1 shows a diagram of the SAM-Grid.

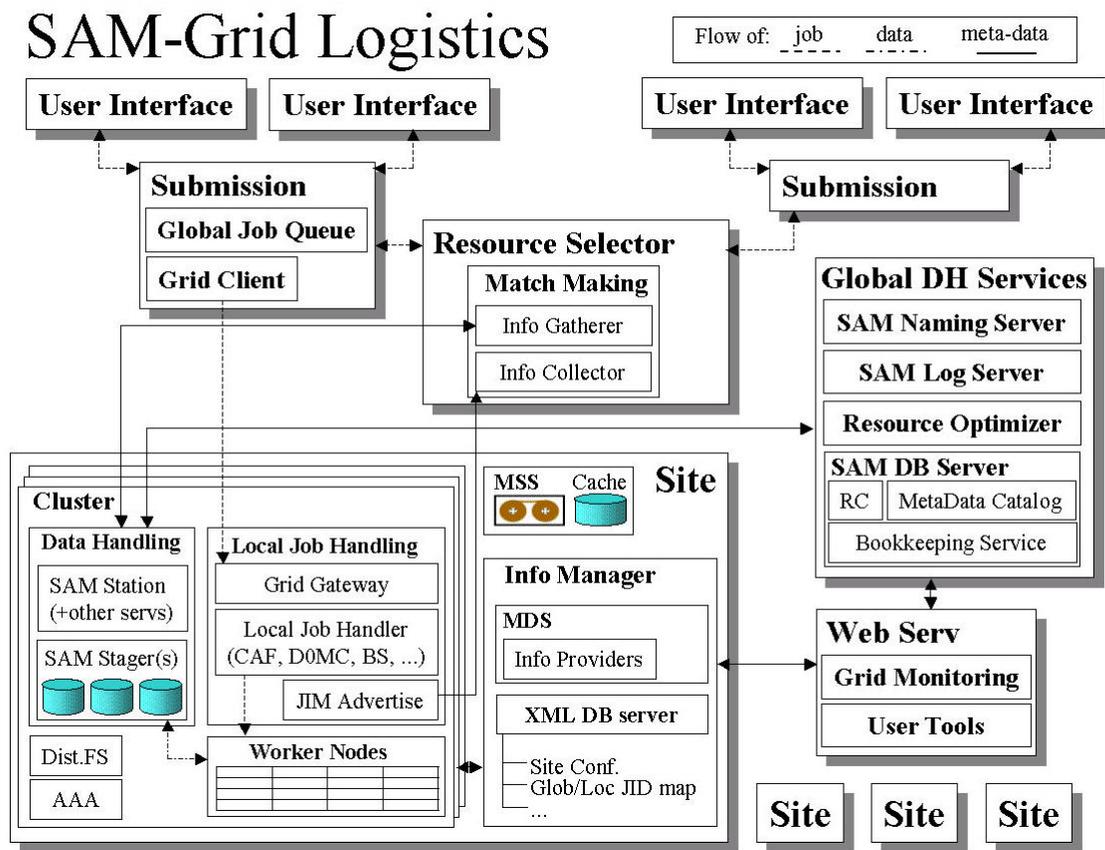


Figure 1: The SAM-Grid services and their multiplicity. Dashed lines represent job flow, solid lines meta-data flow, dash-dotted lines data flow.

## The JIM V1 security model

The SAM-Grid integrates several standard Grid components, such as the Globus Toolkit and Condor-G: the SAM-Grid therefore inherently uses X509 certificates as the primary way for authentication. GSI allows mutual client-server authentication/authorization: first (authentication) certificates are checked against the list of trusted CA where the client/server run; second (authorization), the certificate subject can optionally be checked

<sup>2</sup> JIM is currently interfaced with the D0MC, CAF and SAM-Submit frameworks.

against an identity list or map (e.g. a certificate subject to local UID authorization). For JIM V1, clients do not use identity lists i.e. there is no client side authorization policy on the interaction with an authenticated server. “Official” identity lists, used on the server side for authorization, are maintained centrally and can be pulled periodically at the required sites. Local administrators, though, are ultimately responsible for the maintenance of the authorization lists.

In JIM V1, we distinguish between certificates issued to people and to services. We stress this distinction in the following description of the functionalities of the SAM-Grid services; we argue about the expectations for mutual trust and, what type of information needs to be protected by the various services.

## **Client Site**

The client software is used to submit the user’s job to a submission site, monitor its status, modify its description or cancel it.

The client expects the submission site to allocate and maintain the amount of disk space necessary to hold the compressed input sandbox of the job and the delegated<sup>3</sup> user’s proxy. It also expects the submission site to act on the user’s behalf when submitting the job to the execution site and when retrieving its status, stdout, stderr. The client expects the submission site to protect the input sandbox against alteration and the delegated user proxy against disclosure.

The information flow during submission is the following: the client contacts the resource selector and obtains the physical address of the submission site<sup>4</sup>; it then contacts the submission site to delegate the job submission. The client needs to trust the CA that signed the certificate of both services. No identity list is maintained at this site.

## **Submission Site**

The submission site is responsible for accepting a job from a client, for keeping a queue of the jobs and for reliably submitting each of them to the execution site selected by the resource selector.

The submission site expects the resource selector to return the address of an execution site capable of running the job. If this address is wrong or belongs to an execution site not trusted by the submission site, the job submission fails and the resource selector is asked to provide a different execution site address. Since the submission and execution sites are ultimately responsible to establish a mutual security context, the security of the communication between the submission site and the resource selector is not crucial.

The submission site expects the execution site to locally queue the job, execute it, and report its status and output/error streams. It expects the execution site to guard the job and the output/error streams from alteration and the delegated user proxy from disclosure.

---

<sup>3</sup> The current version of condor does not delegate the proxy to the submission site; it copies the user’s proxy instead. This behavior is going to be changed in future versions of condor to include various certificate retrieval mechanisms, like MyProxy.

<sup>4</sup> Note that here the resource selector is acting as a naming service for its registered services.

The submission site needs to trust the CA that signed the service certificate of the resource selector, the execution site gateway (globus gatekeeper) and the CA that signed the user certificate used by the client. It also needs to maintain an authorization list for such users, since they are the primary beneficiaries of its services/resources. The submission site daemon is called `condor_schedd` and runs under the `sam` UID.

## Execution Site

The execution site is responsible for accepting jobs from the submission site; for advertising itself to the resource selector and for transferring the input files required by the jobs (via SAM).

The servers running at the site are the following:

- **Globus Gatekeeper:** it is the server that receives the requests for scheduling a job by a submission site on behalf of the user. It needs to trust the CA that signed the user's certificate and it needs to keep an authorization list (`grid-mapfile`) of the users authorized to run at the local resource. The server runs as root and its identity can be the host certificate.
- **gridftp:** we run the `gridftp` daemon to enable external access to the files cached by the local SAM station. The daemon runs as UID `sam` under a `sam` service certificate, whose subject contains the machine node name. Each `gridftp` has access to the list of `sam` service subjects that are part of the SAM Grid: this mechanism protects against unauthorized use of the local disk space.
- **jim\_advertise:** it is the service that advertises resources to the Resource Selector. The resource description does not need a high level of protection, since the Resource selector uses it only to *recommend* an execution site. Administrators can choose to run `jim_advertise` under a dedicated service identity or, as we recommend for JIM V1, the same certificate used by `gridftp`. The server will run under the `sam` UID.

## Resource Selector

It is responsible to match jobs with resources and it acts as a naming service for the clients to find the address of the submission sites. It maintains a list of submission and execution sites authorized to register with it, to avoid information flooding from unauthorized services. It will run under a service certificate as the `sam` UID.

## Trusted CA and maintenance of the identity lists

The SAM-Grid will recommend that the site administrators express trust to (possibly a subset) of the European Data Grid CAs and the Fermilab Kerberos CA (KCA). Ultimately, the choice is left to the local administrators.

## Issues related to the FNAL security policy

According to the FNAL policy, only users presenting KCA certificates will be authorized to run jobs<sup>5</sup> at FNAL. Privilege to write files to FNAL on areas dedicated to data will be

---

<sup>5</sup> Jobs that run an executable preinstalled at Fermilab, can also present a certificate signed by an EDG CA.

granted presenting a certificate signed by any of the EDG CA. Privilege to read files will be granted presenting any certificate.

Fermilab KCA grants X509 certificates to users that hold a valid FNAL.GOV kerberos ticket. Note that all of the CDF and most of the DZero collaborators have a Fermilab principal: therefore SAM Grid client installations that have a kerberos client can produce KCA certificates for its users. Users will not be discouraged from obtaining personal certificates from any of the EDG CA, if they want to run their jobs outside Fermilab. Any EDG CA and the KCA can sign Service certificates<sup>6</sup>.

### **Proposed solution to maintain the identity lists**

The identity list of the users of the CDF and DZero VOs can be obtained via the central SAM DB; users of the SAM-Grid will be requested to register to the SAM system. This identity list can be refreshed daily at Fermilab and securely transported using the sam tools (sam\_gsi\_config) to the Submission and the Execution Sites.

The list of sam services (identity list of jim\_advertise and gridftp) can be maintained centrally at Fermilab and pulled by the Execution sites to authorize gridftp connections and by the Resource Selector site to authorize jim\_advertise connections.

The list of Submission Sites can also be maintained centrally and pulled by the Resource Selector site.

Note that the administrators have the ultimate responsibility to maintain this authorization lists and we are willing to cooperate with the integration with other mechanisms to maintain such lists.

---

<sup>6</sup> For JIM V1, we have to change the implementation of some services e.g. sam\_gridftp, which currently assumes that the sam service certificate is issued by the KCA.