Secure Grid Monitoring, a Web-based Framework

Brajendra K. Singh, Amirhasan Amintabar, Akshai Aggarwal, Robert D. Kent,
Ahmedur Rahman, Farhan Mirza, Zillur Rahman
School of Computer Science
University of Windsor, Ontario, CANADA
{Singh11g, amintab, akshaia, rkent, rahma21, mirzad, rahma26} @uwindsor.ca

ABSTRACT
The problem of securely monitoring the grid, in which a group of different entities provide and exchanging confidential information has become a significant task for an efficient use of shared resources. In this paper, a Web based secure grid monitoring framework is presented which permits resources to be monitored only by authorized users. The proposed framework provides monitoring a grid environment comprehensively, with low overhead for authentication and authorization purposes. Another advantage of our framework is that complex authorization policies for grid monitoring can be easily applied.

Keywords
Secure Grid Monitoring, Authentication, Authorization, CAS, Directory Service, SAML.

1. INTRODUCTION
Grid computing uses a dynamic set of available geographically distributed heterogeneous resources, connected by a network, to solve massive computational/ scientific problems. The huge computational power of the grid relies on shared resources which are usually administrated by local domains. Some resources may go offline and leave the grid at any time; resources with new sharing policies may join at any moment; new applications come online, engage some resources and then terminate and release the resources; software or hardware crashes can happen and network congestions are also likely to occur.

Due to this dynamicity and heterogeneity of a grid, the resources need to be monitored and different types of alerts are required to be made available to the grid entities. In other words, grid monitoring helps with performance analysis, fault detection and recovery mechanisms. Some requirements of an ideal grid monitoring system include scalability, high extensibility, timely information updates, low monitoring overheads and security.

GridNets 2007 October 17-19, 2007, Lyon, France.
 ISBN 978-963-9799-07-3.
DOI 10.4108/gridnets.2007.2170

A comparative study of various grid monitoring tools is given in [1]. These tools are specialized for specific measurement scenarios. For example, Ganglia [2] monitors system's performance metrics such as memory used, bytes in, bytes out, available disk spaces, CPU load and network load. Another monitoring tool, MonALISA [3] is based on a scalable Dynamic Distributed Services Architecture which is implemented using JINI/JAVA and WSDL/SOAP technologies [4]. It can monitor the status of each site and produce global statistical data for activities such as system utilization and jobs running at different centers. GridCat[5] is another system monitoring tool which displays available CPU slots and disk information dynamically. GridICE[6] is a grid service monitoring tool, which uses the Grid Laboratory Uniform Environment (GLUE) schema [7] as the common information model. Glue scheme parameters have been defined by Glue Schema Working Group [32] as a set of attributes and attribute semantics to facilitate and standardize the interoperation between Grid infrastructures. NetLogger [8], [9] combines network, host and grid application events and thus provides an overall view that facilitates the identification of performance bottlenecks. NetSaint [10] is another network monitoring tool, which can be used in a grid to view a snapshot of the status of the grid resources and the network. It can alert the entities about the availability of various grid services. WebMDS [12] makes use of the GLUE schema and integrates and provides information from monitoring tools such as Ganglia and Hawkeye. However, WebMDS is still weak on the authorization front.

Our study shows that the current grid monitoring tools have the following problems:

1) Despite some efforts such as GRID3 monitoring system [11] that integrates various Grid monitoring tools, there is no unified system or tool that can provide information metrics related to all the parameters defined in GLUE schema in the same interface.

2) With some exceptions such as in MonALISA [3] that provides ACL based authorization, there is a lack of mechanism for enforcing authorization policies in grid monitoring systems.

3) No framework exists, which addresses the implementation of the Web based secure grid monitoring in full scale.

A *Web based secure monitoring* framework is presented in this work to address the above problems. *Secure grid monitoring* can be defined as the process of gathering information related to grid components and making it available to the permitted users only. One of the issues in the secure grid monitoring is that the existing

enterprises and groups use different security protocols in applying authentication and authorization mechanisms. Accordingly, these policies need interpretation by other administrative domains. Another point is that authentication mechanism is often used in grid monitoring systems [12] [1], however authorization is still an emerging area of research in secure grid monitoring. Unauthorized monitoring can be as dangerous as unauthenticated monitoring. It may expose the vulnerabilities of the particular grid environment, if any, to an unauthorized grid user. An advantage of full scale implementation of authorization in monitoring is to group users in different classes of access rights depending upon factors such as pricing policies and management hierarchy. Based on these authorization policies, a grid user can be authorized to monitor a single or many metrics.

Grid Security Infrastructure (GSI) defines the security requirements for a grid. It uses public key infrastructure for authentication purposes. Globus Toolkit [26] supports GSI. It uses X.509 based proxy certificate concept for authenticating the grid user as well as for delegation of user credentials. For authorization, the services such as CAS [13, 14], VOMS [15], Akenti [16] and PERMIS [17] are available. CAS gives authorization in terms of access rights of the grid user. It can define fine grained authorization policies. In VOMS, a grid user's access rights are defined by the user's membership of a user group. It is suitable for defining coarse grained authorization policies. For both CAS and VOMS, authorization assertions need interpretation at the grid resources. Akenti and PERMIS are more suited for Web Services. In addition to grid security, web transactions also need to be secured. For this purpose, various web service security standards such as WS-Security [18], WS-Trust [19] and WS-Secure Conversation [20] are available. Security Assertion Markup Language (SAML) [21, 22, and 23] is an XML based framework for making statements about the security scenario. It is defined in terms of assertions, protocols, bindings, and profiles. SAML Assertions includes authentication, attribute and authorization information about the entity involved in the web transaction.

In this paper we propose a Web based secure grid monitoring framework. In our approach we introduce an intermediary web service, which is called Secure Monitoring Service (SMS) throughout this paper. In our frame work we take the advantage of using multiple grid monitoring tools for comprehensive grid monitoring. In the proposed framework, CAS server is applied for authorization policy management. We also present the format of SAML security context token that embeds CAS authorization policy assertions into SAML authorization assertions. We discuss the Web based forward trust [25] approach and its applicability in multiple grid monitoring tool scenarios. An implementation of our proposed framework is also presented.

The rest of the paper is organized as follows: Section 2 defines our goals and issues; Section 3 presents the detailed analysis of the problem. Section 4 presents the mapping of CAS authorization assertions onto SAML-SCT. Section 5 discusses the web based forward trust approach for grid monitoring. We present our new framework in Section 6. Section 7 highlights the current status of our implementation of the framework. Conclusion and future scope is presented in the last Section.

2. GOALS AND ISSUES

The goal of this paper is to obtain a web based secure grid monitoring with the following two objectives:

1. Minimum overhead for authorization and authentication.
2. Scalable support for the maximum number of grid applications and grid resource parameter monitoring

The issues which are involved in this work are as follows:

1. Web based interaction with a grid is needed. Therefore, both web security standards and grid security need to be satisfied.
2. Several grid monitoring tools are required to cover all the Glue schema [7] monitoring parameters.
3. An interface is needed between a grid user and the grid monitoring tools.
4. Some authorization mechanism is needed to ensure authorized monitoring of resources.

3. PROBLEM ANALYSIS

Some of the grid monitoring tools [1], [2], were designed for monitoring clusters or distributed environments. To make them meet grid monitoring requirements either the tools were upgraded or some work-around was devised. As a result, the tools do not support the monitoring of all the grid parameters defined by the GLUE schema [7]. Even the monitoring tools that were designed specifically for grid monitoring purposes, such as MonALISA [3], do not monitor all the grid parameters of the GLUE schema. Therefore, to monitor all the grid parameters, defined in the GLUE schema, one has to use more that one monitoring tool. Another problem is that with a few exceptions [3], most of the known monitoring tools do not support the grid security requirements of both authentication and authorization. Some monitoring tools [1] support the Grid Security Infrastructure (GSI); however, the focus is primarily on the authentication part of the security, and the tools do not have any mechanism to handle authorization. MonaLISA provides Access Control List (ACL) based authorization; but, it is not comprehensive in terms of giving all the GLUE schema parameters and it is not scalable in terms of number of entities being served or nodes being monitored at the same time. Moreover its JAVA implementation puts some limits on its performance for a wide spread deployment on grid [1].

Thus, to monitor a grid comprehensively, we need to use more than one monitoring tool simultaneously [11]. In this case, there is a need for a Result Integration Module to combine the results obtained by various tools and presents them to users in a unified and unambiguous manner. In our approach, we emphasize grid parameters to be composed as per the GLUE schema.

Due to lack of authorization functionality in most of the existing grid monitoring tools, in the introduced framework, we have provided an intermediate *Authorization Module*. It would then take care of authorization processes on behalf of the monitoring tools which are engaged. There are three advantages of having such a separate Authorization Module:

First no modifications need be done on existing monitoring tools to support the authorization policy enforcement.

Second, one issue of authorization from the Authorization Module is enough to receive data collected from even a multiple monitoring tools. If a grid user's requested monitoring parameter

involves data taken from more than one tool, he/she need not to prove his/her authorization to each of the services that runs the corresponding monitoring tool. Thus if a grid user is once authorized by the Authorization Module, then she can get monitoring data, for which he/she is authorized, irrespective of the monitoring tool used to collect that data.

Third, any complex authorization policy pattern could be implemented without worrying about whether the monitoring tool would be able to interpret such an authorization policy or not, since the task of interpretation would be done by the Authorization Module for once and enough for all.

Another aspect of the grid monitoring problem is that the geographic location of the grid user with respect to the location of the grid resources can vary widely. Therefore, there is a need to monitor a grid remotely by using the web based solutions. However, a common approach for embedding security in grids is offered by Grid Security Infrastructure (GSI) for security [24]. In this respect, Web systems have their own security related specifications such as WS-Security [18], WS-Secure Conversation [20], and WS-Trust [19] etc. In a secure Web based monitoring solution we believe that GSI and Web service security are required to work together, and it is considered in the proposed framework.

In the proposed framework a web service is used to provide both authorization and result integration functionality. We call this web service the Secure Monitoring Service (SMS) as introduced in Section 1. In addition to authorization and result integration functionality, SMS provides authentication functionality as well. These functionalities of SMS are depicted in Figure 1. Advantage of having a separate web service dealing with security issues is that one time issuance of the authorization certificate suffices accessing multiple tools according to the rights defined. This, consequently, reduces the authorization and authentication overhead in the multi-tool grid monitoring scenario on the whole system. Another advantage of having a separate web service is that SAML [21] security context tokens (SAML-SCT) [25] can be interpreted easily by a web service rather than by GSI based grid middleware. In general, Globus Toolkit is used widely as a grid middleware implementation. Its current version i.e. Globus 4.0.3 [26], [27] supports the WS-secure conversation compliant secure conversation only between the trusted entities. It has its own security context token and it uses SSL/TLS protocol for negotiation to establish connections between the entities involved. Thus, it would not understand SAML-SCT, which comes through web service interaction. As mentioned earlier in our framework, SMS is a web service so it can understand SAML-SCT directly. Thus by using SMS as an intermediary service, there will be no need of a parser or wrapper program to convert SAML-SCT into Globus security context tokens.

4. MAPPING OF CAS AUTHORIZATION ASSERTIONS ONTO SAML-SCT

Wang [25] has proposed a web based secure conversation establishment protocol that uses forwarded trust relationships. In this protocol, SAML [21] Authentication assertions are used to encapsulate the conversation context as well as the conversation target identity authentication information into a Secure Context Token (SCT) called SAML-SCT. Security token and context are two parts of SCT. The security token is used to ensure the confidentiality and integrity of the messages. In PKI (public key infrastructure) implementation, as employed by the Globus Toolkit, a security token contains identity certificates signed by a CA with a predefined life time. The context defines the conversation itself. It gives the information about the life time of the conversation as well as the conversation identity

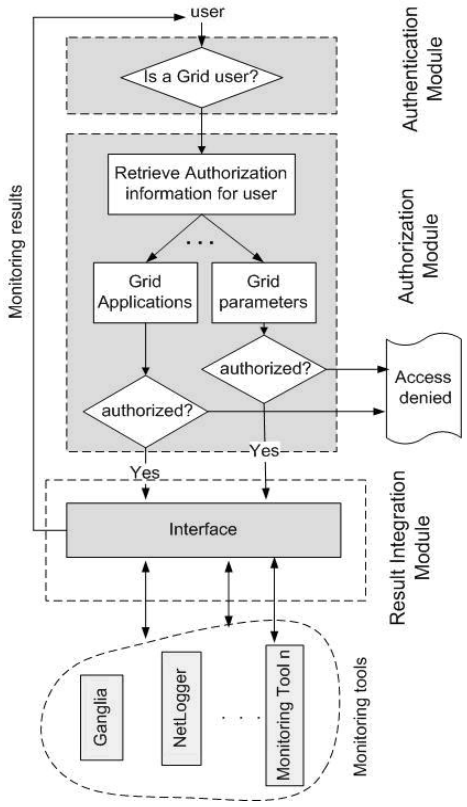

Figure 1. Structure of Secure Monitoring Service (SMS)

The author in [25] has presented the mapping of SCT into SAML-SCT for the authentication assertion part with identity information given in X.509 format. However, in our approach authorization is also applied. We use SAML-SCT [21] to embed CAS authorization policy assertions [14]. The modified SAML-SCT is given in Figure 2. The lower half of Figure 2 highlights the authorization assertions part of SAML-SCT for accessing a grid resource for monitoring purposes. The CAS authorization assertions are mapped onto SAML-SCT. The *Name Identifier* field gives the information about the subject, i.e. the grid user who initiated the request. This user belongs to a CAS user group. The *resource URI* field is used to convey the information about the grid, which here is represented as a CAS object.

CAS issues authorization assertions for a grid user on this grid resource. The *'Decision'* field gives the information about the decision by the CAS about the user regarding the specified grid resource, in terms of 'permission granted or denied'. The *'Action'* field defines one or more action(s), permitted by CAS, to a user about the CAS object. An action is associated with *'ServiceType'* in CAS. For example *'ServiceType'* can be 'file' and the relevant action can be 'Read or Write'. These actions and associated service types can be used to frame complex authorization patterns.

The '*Evidence'* field is optional and gives information about the set of assertions that CAS relies on while making the decisions.

The CAS authorization assertions for a grid user are embedded into SAML-SCT, issued for conversation target only. These authorization assertions need not be embedded into SAML-SCT for grid users as they are intended to be used by the grid resource side only. This means that SAML-SCT for each grid user will contain only authentication information of the conversation target. Thus SAML-SCT for grid user will remain the same as proposed by Wang [25].

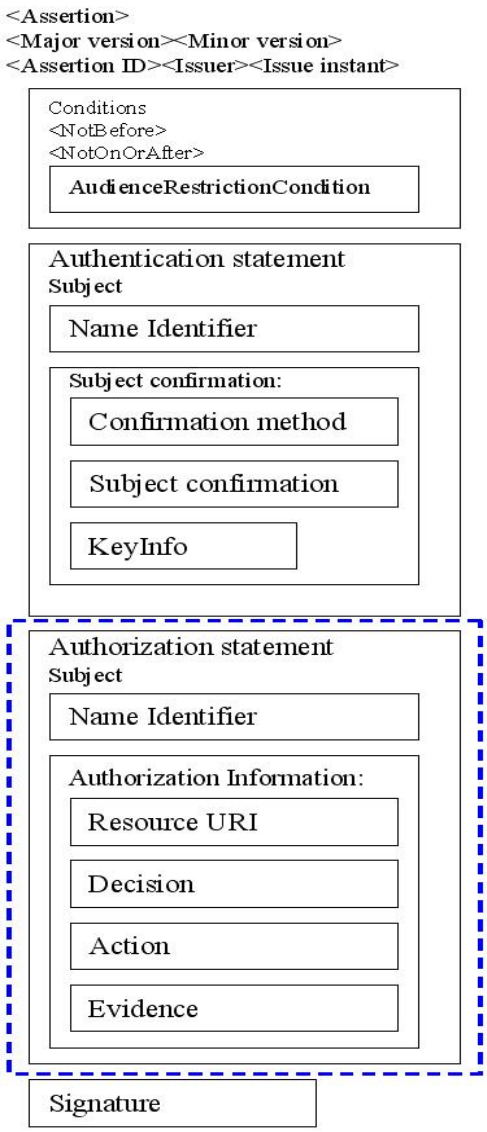

Figure 2. SAML-SCT Token format for a Grid Resource

The latest version of CAS is providing its policy assertions in SAML format [13], therefore it becomes easy to embed CAS authorization assertions into SAML-SCT. Several projects like OGSA-DAI [28] are already using CAS authorization assertions in SAML format.

5. WEB BASED FORWARD TRUST APPROACH FOR GRID MONITORING

For web based grid monitoring, we used issue-forward-use approach [25] for establishing a trust relationship between a grid user and the grid resource, as shown in Figure 3. It is an Indirect Trust relationship based Secure Conversation (ITSC) as the grid user and grid resources depend on a third party (a 'Directory' service) to establish trust relationship as well as to secure conversation among them. A grid user connects to a directory service to retrieve her identity information (such as a CA signed X.509 identity certificate in case of Globus) as well as grid resource identity information (such as a CA signed host/service X.509 identity certificate). The grid user then keeps the grid resource identity information with her and forwards her identity information to the grid resource for authentication. Thus, this approach is also known as the 'Forward Trust' approach. Now, if both the grid user and grid resource to trust the directory service, they can authenticate each other and can start a secure conversation.

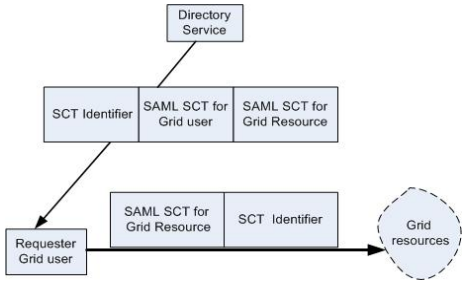

Figure 3. Forward trust scenario

There are two advantages in using issue-forward-use scenario for secure grid monitoring. First, there is no need for the grid resource side to connect to the directory service to retrieve or verify the identity of the grid user as it is able to get the identity information directly from the grid user in the form of an identity certificate embedded into SAML-SCT. This reduces the burden of the grid resource for security confirmation with the directory services. Second, as a result of reducing the direct interaction between the grid resource and the directory service; a large number of interactions are avoided, recalling that the number of resources to be monitored in a grid could be very large. Thus, it reduces the possibility of denial of service due to the overload caused by the interactions with directory service.

6. THE WEB BASED FRAMEWORK

This section introduces the web based secure monitoring framework by giving the definition of the terminologies first, then assumptions and finally presenting the architecture.

6.1 Terminology

Some technical terms used in this framework are explained first.

Grid user: A grid user has a valid user certificate from a certification authority to use grid services.

Directory Service: A directory service lists all the services available within the grid environment.

Community Authorization Service: A Community Authorization Service (CAS) is used to provide authorization assertion that gives information about rights of a Grid user in accessing the resources.

6.2 Assumptions

In this work, it is assumed that the grid user and SMS are in trust with directory service, either directly or through a federated approach, such as WS-Federation and bridged CAs[29]. CAS and MyProxy server [30] are assumed to be in trust with the directory service. The Monitoring tool servers are also assumed to be in trust with SMS.

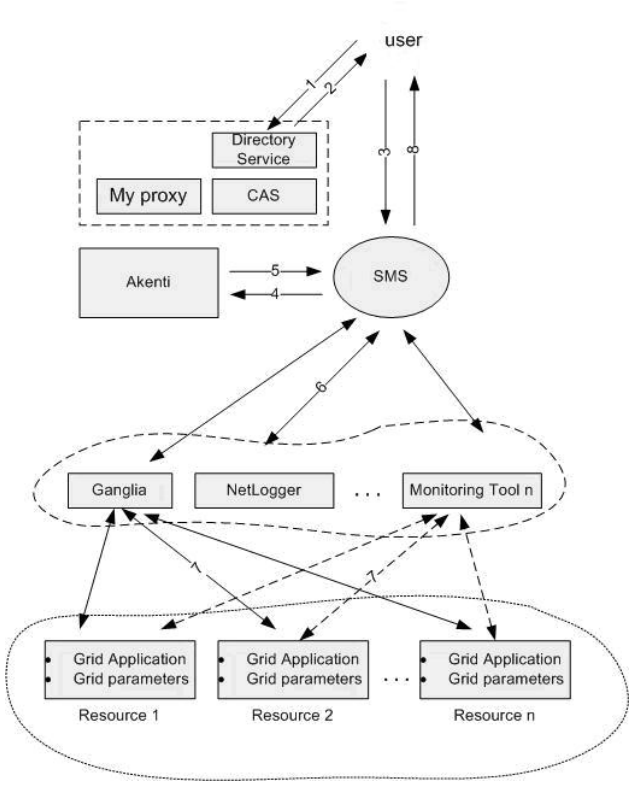

Figure 4. The framework

6.3 Framework

Figure 4 illustrates the architecture of our proposed framework. A grid user makes a request to the directory service to connect to the SMS. The directory service then validates the grid user, based on the username and password. After validation, the directory service interacts with MyProxy server for user identity certificates and with CAS server for authorization assertions. The directory service will then provide the SAML-SCT for the grid user, SAML-SCT for SMS service and SCT context identifier. SAML-SCT for the grid user contains identity certificates of SMS. SAML-SCT for SMS contains identity certificates of grid user as well as CAS authorization assertion. Then, the directory service forwards SCT context identifier and both the SAML-SCTs to the grid user. The grid user keeps the SAML-SCT intended for her and forwards SCT identifier and SAML-SCT intended for SMS to the SMS. SMS authenticates the grid user based on its identity certificate. After this, SMS forwards the grid user's authentication information as well as the CAS assertions to Akenti [13, 16]. Akenti is capable of parsing complex authorization decisions. It interprets these authorization assertions and gives the authorization decision to the SMS. Now SMS interacts with the individual monitoring services, like Ganglia and Netlogger. These monitoring services collect the information from the grid resources in terms of grid parameters. The Netlogger service, through its API calls, collects the information about the grid application running on the grid resources. Thus, status of the grid application and the values of the grid parameters become available to the SMS. Now, on the basis of authorization rights of the grid user, SMS generates a response packet by integrating relevant monitoring result about the grid applications and grid parameters. This response packet is sent back to the grid user. The grid user and SMS will remain in secure conversation as long as the context is valid.

The proposed framework is totally conformant with WS-Security [18], WS-Trust [19], WS-Secure Conversation [20] and WS-Security SAML token profile [31].

7. IMPLEMENTATION

To provide our concept, we installed Globus Toolkit version 4.0.3 on six computers and six Grid users were then created. Every grid user gets a user certificate signed by the CA [26]. We installed Ganglia version 3.0.4 and Netlogger version 3.3.11. We used SAML version 1.1 for web interaction of the grid users with the proposed SMS. CAS server version 2.0 was installed and a grid user called CAS administrator was created to manage who is other CAS users and CAS object. The CAS service is accessible to other CAS user through the web address http://hostname1:8080/wsrf/services/CASService. Here the hostname1 represents the host name of the system where the CAS server is running. We then created four CAS groups namely Researcher, Resource provider, Resource user and Guest. Then, we enrolled six grid users to these CAS groups depending upon their roles in the Virtual Organization (VO). This enrollment process makes them valid CAS users. We created three CAS object groups namely Grid resource, Grid monitoring parameter and Grid application. Then, we added six computers: the grid monitoring parameters such as GlueHostName, GlueHostProcessorLoad and GlueHostMainMemory and the user defined applications such as large matrix manipulation program and load balancing servers to these CAS object groups in the suitable categories. We also created actions namely view, modify, and view & modify. The permissions are granted on these actions to the CAS user groups that basically creates the CAS Authorization policies in the grid monitoring system. The CAS authorization policy enforcement is done by wrapping Globus commands into CAS wrapper using cas-wrap command.

8. CONCLUSION

In this paper we have analyzed the requirements of a web based secure grid monitoring system. We found that a web service, which would act as an intermediary between the grid user and the grid monitoring system, is needed. We mentioned that a CAS server is needed for enforcing authorization mechanism in the grid

monitoring system. We also determined that, given the current capability status of available grid monitoring tools, it is necessary to use several grid monitoring tools to facilitate comprehensive monitoring of grid applications and grid parameters. Thus, based on CAS, an intermediary web service and SAML, we have proposed a framework for web based secure monitoring. We have carried out an extensive feasibility study and a concept proving implementation of our proposed framework.

In future, several other monitoring tools will be integrated with our proposed framework. We will also enhance the capability of SMS to convert monitored data, collected from various monitoring tools, into grid parameters as per GLUE schema. We shall also study the possibility of enhancing the capability of authorization module of SMS to understand complex CAS authorization assertions.

9. REFRENCES

[1] Zanikolas, S., Sakellariou, R. 2005. A Taxonomy of Grid Monitoring System. Future Generation Computer Systems. vol 21, issue 1 (January 2005).

[2] Massie, M.L., Chun, B.N., Culler, D.E. 2004. Ganglia Distributed Monitoring System: Design, Implementation, and Experience. Parallel Computing 30, pp. 817–840.

[3] Newman, H.B., Legrand, I.C., Galvez, P., Voicu, R., Cirstoiu, C. 2003. MonALISA: a distributed monitoring service architecture. Computing in High Energy and Nuclear Physics (CHEP03). (La Jolla, CA, 2003).

[4] Web Services Description Language (WSDL) Version 2.0, SOAP Specifications, World Wide Web Consortium, http://w3.org

[5] GridCat. 2007. http://www.ivdgl.org/gridcat/home/ (accessed Jan 2007).

[6] Andreozzi, S., Bortoli, N. D., Fantinel, S., Ghiselli, A., Tortone, G., Vistoli, C. 2003. GridICE, A Monitoring Service for the Grid. Proceedings of the Third Cracow Grid Workshop, Cracow. (Poland, October 27–29, 2003). pp. 220–226.

[7] GLUE Schema Specification version 1.2: http://glueschema.forge.cnaf.infn.it/uploads/Spec/GLUEInfoModel_1_2_final.pdf (as of Dec 2006)

[8] Gunter, D., Tierney, B., Crowley, B., Holding, M., Lee, J. 2000. NetLogger: a toolkit for distributed system performance analysis. Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings. 8th International Symposium. (29 Aug.-1 Sept. 2000) pp. 267 – 273.

[9] NetLogger, http://dsd.lbl.gov/NetLogger/overview.html

[10] Barbera, R., Re, P.L., Sava, G., Tortone, G. 2002. Grid monitoring with NetSaint. Bologna-Datagrid WP7 meeting, (Jan 24, 2002).

[11] Mambelli, M., Kim, B., et al. 2004. Grid2003 Monitoring, Metrics and Grid Cataloging System. Proc. of CHEP04, Interlaken, (CH, Sept 2004).

[12] GT 4.0 WS MDS WebMDS: http://www.globus.org/toolkit/docs/4.0/info/webmds/

[13] Foster, I., Kesselman, C., Pearlman, L., Tuecke, S. and Welch, V. 2003. The Community Authorization Service: Status and Future. Proceedings of Computing in High Energy Physics 03 (CHEP '03). (La Jolla, California, March 24-28, 2003).

[14] GT4 CAS Admin Guide, http://www.globus.org/toolkit/docs/4.0/security/cas/admin-index.html

[15] Alfieri, R., Cecchini, R., Ciaschini, V., dell'Agnello, L., Frohner, A., Gianoli, A., Lorentey, K. and Spataro, F. 2003. VOMS, an authorization system for virtual organizations. European Across Grids Conference. (2003), pp. 33–40.

[16] The Akenti Approach, http://dsd.lbl.gov/Akenti/

[17] Chadwick, D. W. and Otenko, A. The PERMIS X.509 Role Based Privilege Management Infrastructure. 7th ACM Symposium on Access Control Models and Technologies. (2002).

[18] WS-Security Standard V1.0, OASIS Web Services Security TC. 2004. http://docs.oasis-open.org/wss/2004/01/oasis-200401-wsssoap-message-security-1.0.pdf

[19] WS-Trust V1.0 Working Draft, OASIS Web Services Secure Exchange TC. 2006. http://www.oasisopen.org/committees/download.php/16138/oasis-wssx-ws-trust-1.0.pdf

[20] WS-SecureConversation V1.0 Working Draft, OASIS Web Services Secure Exchange TC. 2006. http://www.oasisopen.org/committees/download.php/16140/oasis-wssx-wssecureconversation-1.0.pdf (as of Nov. 2006).

[21] Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1, OASIS Standard. 2003. http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf . (2 September, 2003).

[22] Technical Overview of the OASIS Security Assertion Markup Language (SAML) V1.1. 2007. http://www.oasis-open.org/committees/download.php/6837/sstc-saml-tech-overview-1.1-cd.pdf. (January 9, 2007)

[23] Security Assertion Markup Language http://www.oasisopen.org/committees/tc_home.php?wg_abbrev=security

[24] Welch, V., Siebenlist, F., Foster, I., Bresnahan, J., Czajkowski, K., Gawor, J., Kesselman, C., Meder, S., Pearlman, L., Tuecke, S. 2003. Security for Grid Services. Twelfth International Symposium on High Performance Distributed Computing (HPDC-12), IEEE Press. (June 2003).

[25] Wang, J. 2006. A Web Services Secure Conversation Establishment Protocol Based on Forwarded Trust. Web Services, ICWS '06. International Conference. (Sept. 2006) pp. 569 – 576

[26] Foster, I. 2006. Globus Toolkit Version 4: Software for Service-Oriented Systems. IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, pp 2-13.

[27] Globus, http://www.globus.org/ (accessed 9th January 2007)

[28] Pereira, A. L., Muppavarapu, V. and Chung, S. M. 2006. Managing Role-Based Access Control Policies for Grid

Databases in OGSA-DAI Using CAS. Journal of Grid Computing. Springer. (Dec 2006), pp. 17.

[29] Jokl, J., Basney, J. and Humphrey, M. 2004. Experiences using Bridge CAs for Grids, Proceedings of the UK Workshop on Grid Security Experiences Oxford. (8th - 9th July 2004).

[30] Novotny, J., Tuecke, S. and Welch, V. 2001. An Online Credential Repository for the Grid: MyProxy. Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10), IEEE Press.

[31] WS-Security SAML Token Profile V1.0, OASIS Web Services Security TC. 2004. http://docs.oasis-open.org/wss/oasis-wsssaml-token-profile-1.0.pdf.

[32] https://forge.gridforum.org/sf/projects/glue-wg (accessed 28th August 2007)