# Survey of Security in Grid Services

**Jianfang Xiao, Dongdai Lin**

The State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences,
Graduate School of the Chinese Academy of Sciences (GSCAS), Beijing 100080, China
Jianfang02@ios.cn

**ABSTRACT**

This article provides a survey of Security in Grid Services coming from a study of many papers most of which were done by the Grid Forum OGSA-SEC (Open Grid Service Architecture Security) working group, GSI (Grid Security Infrastructure) working group, and Globus Alliance team and other people who contributed to Grid. It describes the best practice in terms of Grid Security Challenges, Grid Security Requirements, and the GT3 (Globus Toolkit version 3) Security Model for OGSA. Most of these were further refined in separate documents.

***Keywords*:** Grid, Security, GSI, Globus

## 1. INTRODUCTION

The term "Grid" denotes a proposed distributed computing infrastructure for advanced science and engineering [1, 2]. The Grid Security Infrastructure (GSI) was subsequently developed, based on existing standards, to address the unique security requirements that arise in Grid environments, as described in [3, 4, 5].

Research and development efforts within the Grid community have produced protocols, services, and tools that address the challenges arising when we seek to build scalable *virtual organizations* (VOs). Here a virtual organization is defined as a set of individuals or institutions sharing resources and services under a set of rules and policies governing the extent and conditions for that sharing [6].

Controlling access to services through robust security protocols and security policy is paramount to controlling access to VO resources and assets. Thus, authentication mechanisms are required so that the identity of individuals and services can be established, and service providers must implement authorization mechanisms to enforce policy over how each service can be used. The requirement for composition complicates issues of policy enforcement, as one must be able to apply and enforce policy at all levels of composition and to translate policies between levels of composition [6].

Grid computing research has produced security technologies based not on direct interorganizational trust relationship but rather on the use of the VO as a bridge among the entities participating in a particular community or function. The results of this research have been incorporated into a widely used software system called the Globus Toolkit (GT)[3] that uses public key technologies to address issue of single sign-on, delegation [7], and identity mapping, while supporting standardized APIs such as GSS-API [8].

The recent definition of the Open Grid Services Infrastructure specification and other elements of the Open Grid Services Architecture (OGSA) [9] within the Global Grid Forum introduce new challenges and opportunities for Grid security.

Integration of GSI with OGSA enables the use of Web services techniques to express and publish policy [10], allowing applications to determine automatically what security policies and mechanisms are required of them. Implementing security in the form of OGSA services allows those services to be used as needed by applications to meet these requirements [11].

The remainder of this document is structured as follows. First, the following sections present a review of security challenges encountered in grid environments. Then discuss the security requirements. At last the document survey an OGSA security model and the GT security model.

## 2. SECURITY CHALLENGES IN A GRID ENVIRONMENT [6]

The security challenges faced in a Grid environment can be grouped into three categories. A solution within a given category will often depend on a solution in another category.

### 2.1 The Integration Challenge

It is unreasonable to expect that a single security technology can be defined that will both address all Grid security challenges and be adopted in every hosting environment. Thus, to be successful, a Grid security architecture needs to step up to the challenge of integrating with existing security architectures and models across platforms and hosting environments. This means that the architecture must be *implementation agnostic*, so that it can be instantiated in terms of any existing security mechanisms (e.g., Kerberos, PKI); *extensible*, so that it can incorporate new security

services as they become available; and *integratable* with existing security services.

### 2.2 The Interoperability Challenge

Services that traverse multiple domains and hosting environments need to be able to interact with each other, thus introducing the need for interoperability at multiple levels: At the *protocol level*, we require mechanisms that allow domains to exchange messages. This can be achieved via SOAP/HTTP, for example; At the *policy level*, secure interoperability requires that each party be able to specify any policy it may wish in order to engage in a secure conversation—and that policies expressed by different parties can be made mutually comprehensible; At the *identity level*, we require mechanisms for identifying a user from one domain in another domain. This requirement goes beyond the need to define trust relationships and achieve federation between security mechanisms (e.g., from Kerberos tickets to X.509 certificates).

### 2.3 The Trust Relationship Challenge

Grid service requests can span multiple security domains. Trust relationships among these domains play an important role in the outcome of such end-to-end traversals. A service needs to make its access requirements available to interested client entities, so that they understand how to securely request access to it. Trust establishment may be a one-time activity per session or it may be evaluated dynamically on every request. The dynamic nature of the Grid in some cases can make it impossible to establish trust relationships among sites prior to application execution [4]. Given that the participating domains may have different security infrastructures (e.g., Kerberos, PKI) it is necessary to realize the required trust relationships through some form of federation among the security mechanisms.

The trust relationship problem is made more difficult in a Grid environment by the need to support the dynamic, user-controlled deployment and management of *transient services* [1]. End users create such transient services to perform request-specific tasks, which may involve the execution of user code.

More details about Security Challenges in a Grid Environment occurred in Ref. [6].

## 3. GRID SECURITY REQUIREMENTS [5, 6]

That the goal and purpose of Grid technologies is to support the sharing and coordinated use of diverse resources in dynamic, distributed VOs: in other words, to enable the creation, from distributed components, of virtual computing systems that are sufficiently integrated to deliver desired qualities of service. Security is one of the characteristics of an OGSA-compliant component. The basic requirements of an OGSA security model are that security mechanisms be *pluggable* and *discoverable* by a service requestor from a service description. This functionality then allows a service provider to choose from multiple distributed security architectures supported by multiple different vendors and to plug its preferred one(s) into the infrastructure supporting its Grid services.

OGSA security must be seamless from edge of network to application and data servers, and allow the federation of security mechanisms not only at intermediaries, but also on the platforms that host the services being accessed.

A Grid security solution should be based on existing standards wherever possible. Security is an extremely complex problem, with specific solutions incrementally developed over many years by many extremely talented people. Further, the community generally only trusts a particular security solution if it has stood the tests of time and repeated scrutiny.

However, Grid environments have a broad range of security requirements [3, 4]. Unfortunately, no single, existing, standard security solution addresses all of these requirements, though ideally a Grid security solution would extend existing standards.

Grid authentication requirements include: Single sign on, Delegation, Integration with various local security solutions, User-based trust relationships.

Grid requirements for communication protection include: Flexible message protection, Supports various reliable communication protocols, Supports independent data units (IDU).

Grid authorization requirements include: Authorization by stakeholders, restricted delegation.

Specification about the Security Requirements can be seen in Ref. [5, 6].

## 4. GT2 GRID SECURITY MODEL [11]

We first review briefly the security technologies incorporated in the Globus Toolkit version 2 (GT2) [12] before we survey the OGSA and GT3 security. GT2 includes some services that use a common Grid Security Infrastructure (GSI) [3,4] to provide security functionality.

*Diverse site security mechanisms*. GSI defines a common credential format based on X.509 identity certificates [13,14] and a common protocol based on transport layer security (TLS [15], SSL [16]).

*Dynamic creation of entities and the granting of privileges to those entities*. GSI introduces X.509 proxy

certificates [14] that allow a user to assign dynamically a new X.509 identity to an entity and then delegate some subset of their rights to that identity. Users create a proxy certificate by issuing a new X.509 certificate signed using their own credentials instead of involving a CA. This mechanism allows new credentials and identities to be created quickly without the involvement of a traditional administrator.

*Dynamic creation and management of overlaid trust domains*. The requirement for overlaid trust domains to establish VOs is satisfied by GSI using both proxy certificates and security services such as the Community Authorization Service (CAS) [17]. GSI has an implicit policy that any two entities bearing proxy certificates issued by the same user will inherently trust each other. This policy allows users to create trust domains dynamically by issuing proxy certificates to any services that they wish to collaborate.

In designing GSI we evaluated several related efforts before electing to build on PKI. We noted the following shortcoming in other approaches with respect to Grid security requirements [11]:

Kerberos [18] requires the explicit involvement of site administrators to establish interdomain trust relationships or to create new entities.

The CRISIS wide area security system [19] defines a uniform and scalable security infrastructure for wide area systems but does not address interoperability with local security mechanisms.

Secure Shell (SSH) [20] provides a strong system of authentication and message protection but has no support for translation between different mechanisms or for creation of dynamic entities.

The Legion security model [21] is perhaps the most similar to that of GT2, using X.509 certificates for delegation. However, it lacks mechanisms for creation of dynamic entities.

## 5. THE GT3 SECURITY MODEL FOR OGSA [11]

We now turn to the question of how Grid security challenges can be addressed within the context of the Open Grid Services Architecture (OGSA) [9], a set of technical specifications that align Grid technologies with emerging Web services technologies [16].

Web services technologies allow software components to be defined in terms of access methods, bindings of these methods to specific communication mechanisms, and mechanisms for discovering relevant services. While particular mechanisms and methods are not prescribed, some mechanisms are emerging as ubiquitous.

OGSA defines standard Web service interfaces and behaviors that add to Web services the concepts of stateful services and secure invocation. These interfaces and behaviors define what is called a “Grid service” and allow users to manage the Grid service’s life-cycle, as allowed by policy, and to create sophisticated distributed services. Grid services can define, as part of their interface, service data elements (SDEs) that other entities can (again, subject to policy) query or subscribe to.

OGSA introduces both new opportunities and new challenges for Grid security. Emerging Web services security specifications address the expression of Web service security policy (WS-Policy [10], XACML [22]), standard formats for security token exchange (WS-Security [23], SAML [24]), and standard methods for authentication and establishment of security contexts and trust relationships (WS-SecureConversation [25], WS-trust [26]). These specifications may be exploited to create standard, interoperable methods for these feathers in Grid security. But they may, in some case, also need to be extended to address the Grid security requirements listed above.

GT3 and its accompanying Grid Security Infrastructure (GSI3) provide the first implementation of OGSA mechanisms. GT3’s security model seeks to allow applications and users to operate on the Grid in as seamless and automated a manner as possible. Security mechanisms should not have to be instantiated in an application but instead should be supplied by the surrounding grid infrastructure, allowing the infrastructure to adapt on behalf of the application to meet the applications requirements. The application should need to deal with only application specific policy. GT3 uses the following powerful features of OGSA and Web services security to work toward this goal [11]:

①. Cast security functionality as OGSA services to allow them to be located and used as needed by applications.
②. Used sophisticated hosting environments to handle security for applications and allow security to adapt without having to change the application.
③. Publishes service security policy so that clients can discover dynamically what credentials and mechanisms are needed to establish trust with the service.
④. Specifies standards for the exchange of security tokens to allow for interoperability.

### 5.1 Security as Services

Secure operation in a Grid environment requires that applications and services be capable of supporting a variety of security functionality, such as authentication, authorization, credential conversion, auditing, and delegation. Grid applications need to interact with other applications and services that have a range of security mechanisms and requirements. These mechanisms and

requirements are likely to evolve over time as new mechanisms are developed or policies change. Grid applications must avoid embedding security mechanisms statically in order to adapt to changing requirements.

The OGSA security model casts security functions as OGSA services. This strategy allows well-defined protocols and interfaces to be defined for these services and permits an application to outsource security functionality by using a security services with a particular implementation to fit its current need.

### 5.2 Hosting Environment

It is not a trivial task to fine and use security services such as those described in the preceding section: in fact, it can require considerable sophistication on the part of the application. Ideally, application developers should not be burdened with the details of this process.

Grid services, like the Web services they leverage, may be built on sophisticated container-based hosting environments such as J2EE or .NET. These hosting environments provide a high level of functionality and allow for much security implementation complexity to be pulled from applications. It is envisioned that most security functionality will be placed in hosting environments, simplifying application development and allowing security functionality to be upgraded independently of applications.

### 5.3 Publishing of Security Policy

In order to establish trusts, two entities need to be able to find a common set of security mechanisms that both understand. The use of hosting environments and security services, as described previously, enables OGSA applications and services to adapt dynamically and use different security mechanisms. However, an application can select the proper security mechanisms and credentials only of it know what mechanisms and credentials are acceptable to the service with which it wishes to interact.

The WS-Policy [10] specification and its related specifications define how a Web service can publish its security policy along with its interface specification as part of a WASL document. Such a published policy can express requirements for mechanisms, acceptable trust roots, token formats, and other security parameter. An application wishing to interact with the service can examine this published policy and gather the needed credentials and functionality by contacting appropriate OGSA security services.

### 5.4 Specified format for Security Tokens

The WS-Security [23], WS-SecureConversation [25], and WS-Trust [26] specifications contain conventions and formats for the communication of various mechanism specific tokens (e.g., Kerberos tickets and X.509 certificates) inside SOAP envelopes. This enveloping standardizes the protocol for security mechanisms and allows mechanisms to be independent of any application protocol. Hosting environments can recognize security-related messages and route them to an appropriate service for handling, and entities in the Network can recognize whether and how an interaction is secured.

### 5.5 GT3 Security Advantages

The Grid Security Infrastructure version 3 (GSI 3) of the Globus Toolkit version 3 is an initial implementation of key components of the OGSA security model described above. This implementation has two key advantages over its GT2 predecessor described in Section 4:

① Use of WS-Security protocols and standards. GT3 uses SOAP and the Web services security specifications for all of its communications. This allows it to leverage and use standard current and future Web service tools and software.
② Tight least-privilege model. In contrast to GT2, the GT3 resource management implementation uses no privileged services. All privileged code is contained in two small, tightly constrained setuid programs.

How these two advantages were implemented in GT3 and more details about Section 4, 5 were described in Ref. [11].

## 6. CONCLUSIONS

Besides the human factor (Users may become a threat to the security of the systems sometimes), Grid computing present so many security challenges that building a secure Grid environment is a complex task. The use of security technologies (e.g. policy languages, authorization services, encryption mechanisms) alone does not make for a secure system. GT3 implements the emerging OGSA; its GSI implementation (GSI 3) takes advantage of this evolution to improve on the security model used in earlier versions of the toolkit [11]. Its development provides a basis for a variety of future work.

## ACKNOWLEDGMENT

This paper was supported by the 863 Project (2003AA144030) and NSFC (90204016).

## REFERENCES

[1] Foster, I., C. Kesselman, and S. Tuecke, "*The Anatomy of the Grid: Enabling Scalable Virtual Organizations*," *International Journal of Supercomputer Applications*, 2001.

[2] Stevens, R. and e. al., "*From the I-WAY to the National Technology Grid*," *Communications of the ACM*, vol. 40, pp. 50-61, 1997.

[3] Butler, R., D. Engert, I. Foster, C. Kesselman, and S. Tuecke, "*A National-Scale Authentication Infrastructure*," *IEEE Computer*, vol. 33, pp. 60-66, 2000.

[4] Foster, I., C. Kesselman, G. Tsudik, and S. Tuecke, "*A Security Architecture for Computational Grids*," presented at Proceedings of the 5th ACM Conference on Computer and Communications Security, 1998.

[5] S. Tuecke, *Grid Security Infrastructure (GSI) Roadmap*, http://www.gridforum.org/security/ggf1_2001-03/drafts/draft-ggf-gsi-roadmap-02.doc

[6] Nataraj Nagaratnam, Philippe Janson, John Dayka, Anthony Nadalin, Frank Siebenlist, Von Welch, Steven Tuecke, and Ian Foster, *Security Architecture for Open Grid Services*, http://www.globus.org/ogsa/Security/draft-ggf-ogsa-sec-arch-01.pdf.

[7] Gasser, M. and McDermott, E., *An Architecture for Practical Delegation in a Distributed System.* Proc. 1990 IEEE *Symposium on Research in Security and Privacy*, 1990, IEEE press, 20-30.

[8] Linn, J. *Generic Security service Application Program Interface, Version 2.* INTERNET RFC 2078, 1997.

[9] Foster, I. And Kesselman, C. Nick, J. and Tuecke, S. *The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration*, Globus Project, 2002. http://www.globus.org/research/papers/ogsa.pdf.

[10] BEA, IBM, Microsoft and SAP. *Web Services Policy Language (WS-Policy)*, 2002.

[11] Von Welch, Frank Siebenlist, Ian Foster , John Bresnahan, Karl Czajkowski, Jarek Gawor, Carl Kesselman, Sam Meder, Laura Pearlman, and Steven Tuecke, *Security for Grid Services*, http://www.globus.org/Security/GSI3/GT3-Security-HPDC.pdf

[12] Foster, I. And Kesselman, C. Globus: *A Toolkit-Based Grid Architecture*. Foster, I. And Kesselman, C. eds. *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 1999, 259-278.

[13] CCITT Recommendation X.509: *The Directory – Authentication framework.* 1998.

[14] Tuecke, S.,Engert, D. Foster, I., Thompson, M., Peatlman, L. and Kesselman, C. *Internet X.509 Public Key Infrastructure Proxy Certificate Profile*, IETF, 2001;

[15] Dierks, T.and Allen, C. *The TLS Protocol Version 1.0*, IETF, 1999. http://www.ietf.org/rfc/rfc2246.txt.

[16] Graham, S., Simeonov, S., Boubez, T., Daniels, G., Davis, D., Nakamura, Y. and Neyama, R. *Building Web Services with Java: Making Sense of XML, SOAP, WSDL, and UDDI.* Sams, 2001.

[17] Pearlman, L., Welch, V., Foster, I., Kesselman, C.and Tuecke, S., *A Community Authorization Service for Group Collaboration.* IEEE 3rd International Workshop on policies for Distributed Systems and Networks, 2002;

[18] Neuman, B, C. and Ts'o, T. Kerberos: *An Authentication Service for Computer Networks. IEEE Communications Magazene*, 32(9). 33-88. 1994.

[19] Belani, E., Vahdat, A., Anderson, T. and Dahlin, M. *The CRISE Wide Area Security Architecture*. 8th Usenix UNIX Security Symposium, 1998.

[20] *OpenSSH*, http://www.openssh.com, 2003.

[21] Humphrey, M., Knabe, F., Ferrari, A. and Grimshaw, A., Accountability and Control of Process creation in Metasystems. 2000 *Nettwork and Distributed System Security Symposium*, 2000.

[22] *eXtensible Access Control Markup Language (XACML) 1.0 Specification*, OASIS, February 2003. http://www.oasis-open.org/committees/xacml/

[23] IBM, Microsoft, *RSA Security and VeriSign. Web Services Security Language (WS- Security)*, 2002.

[24] *Security Assertion Markup Language (SAML) 1.0 Specification*, OASIS, November 2002. http://www.oasis-open.org/committees/security/

[25] IBM, Microsoft, *RSA Security and VeriSign. Web Services Secure Conversation Language (WS-SecrureConversation) Version 1.0*, 2002.

[26] IBM, Microsoft, *RSA Security and VeriSign. Web Services Trust Language (WS-Trust)*, 2002.