Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective

The Globus Security Team[1]

Version 2 updated December 8, 2004

Abstract

This document provides an overview of the Grid Security Infrastructure (GSI) contained in Web Services-based components of version 4 of the Globus Toolkit (GT4). Particular attention is paid to the relationship between GT4 GSI components and security standards from GGF, IETF, W3C, and OASIS.

1 Introduction

"Grids" [9] have emerged as a common approach to constructing dynamic, inter-domain, distributed computing and data collaborations. The open source Globus Toolkit® [8] middleware has been developed to support these environments, and is used in Grid deployments worldwide. The Grid Security Infrastructure (GSI) [3, 10] is the portion of the Globus Toolkit that provides the fundamental security services needed to support Grids.

This document describes how GT GSI implements standards from different standards bodies to perform its functions. GT4 contains both Web Service and pre-Web Service Grid components. This document discusses solely the Web Service components. The pre-Web Service components use the same authentication mechanisms as the Web Services components described here, but implement application-specific protocols and message protection schemes that are beyond the scope of this document.

This document does not discuss either the Globus Toolkit GSI components or the standards in detail, only their relationship. Readers interested in more details should read the documents indicated in the references.

2 Transport-level and Message-level Security

GT4.0 supports both message-level and transport-level security, and will continue to support both for the foreseeable future.

By "message-level security" we mean support for the WS-Security standard and the WS-SecureConversation specification to provide message protection for SOAP messages.

By "transport-level security" we mean authentication via TLS[5] with support for X.509 proxy certificates.

---

[1] Document editor: Von Welch <vwelch@ncsa.uiuc.edu>

GT4.0's support for message-level security is important as it allows us to comply with the WS-Interoperability Basic Security Profile. However, because current message-level security implementations have relatively poor performance, GT4.0 services use transport-level security by default. This choice is driven by user performance demands.

The poor performance of message-level security implementations seems to be partly an implementation issue and partly a specification issues, and it is not clear when it will improve.

If and when it does improve, GT can be expected to move to using message-level security as a default. Eventually, transport-level support could be deprecated. However, this would only be after a transition period and will not occur any time soon.

3 GSI Functional Layers

As shown in Figure 1, GSI may be thought of as being composed of four distinct functions: message protection, authentication, delegation, and authorization. Implementations of different standards are used to provide each of these functions:

- TLS (transport-level) or WS-Security and WS-SecureConversation (message-level) are used as message protection mechanisms in combination with SOAP.
- X.509 End Entity Certificates or Username and Password are used as authentication credentials
- X.509 Proxy Certificates and WS-Trust are used for delegation
- SAML assertions are used for authorization

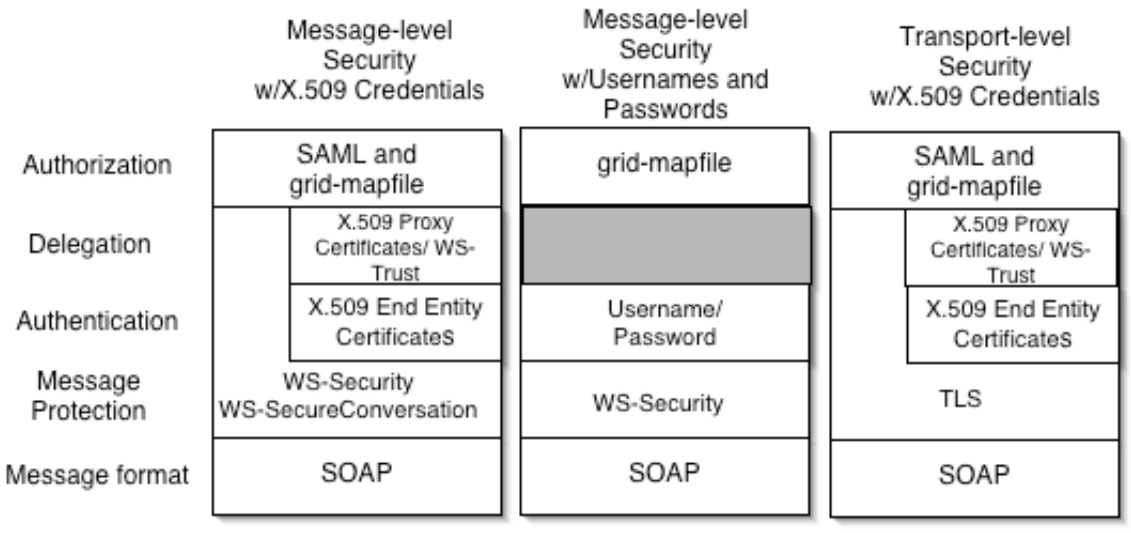

Figure 1: Overview of the GT4 Grid Security Infrastructure and standards used for different functions. The figure on the left shows functionality with X.509 credentials. The figure on the right shows functionality with username/password authentication.

The remainder of this document reviews both the GT implementations of each of these functions and the standards that are used in these implementations. Section 8 discusses the relationship of GSI to WS-I.

4 Message Protection

The Web Services portions of GT4 use SOAP [1] as their message protocol for communication. Message protection can be provided either by transporting SOAP messages over TLS, known as Transport-level security, or by signing and/or encrypting portions of the SOAP message using the WS-Security standard, known as Message-level Security. In this section we describe these two methods.

4.1 Transport-level Security

Transport-level security entails SOAP messages conveyed over a network connection protected by TLS. TLS provides for both integrity protection and privacy (via encryption).

Transport-level security is supported today as a higher-performance alternative to the more standards driven message-level security. If and when message-level security improved in performance, driven by a combination of implementation and specification factors, we expect a gradual depreciation of transport-level security.

Transport-level security is normally used in conjunction with X.509 credentials for authentication, but can also be used without such credentials to provide message protection without authentication, often referred to as “anonymous transport-level security.” In this mode of operation, authentication may be done on a different level, e.g. via username and password in a SOAP message, or communications may be truly unauthenticated.

4.2 Message-level Security

The SOAP specification allows for the abstraction of the application-specific portion of the payload from any security (e.g., digital signature, integrity protection, or encryption) applied to that payload, allowing GSI security to be applied in a consistent manner across SOAP messages for any GT4 Web Service-based application or component.

GSI implements the WS-Security standard and the WS-SecureConversation specification to provide message protection for SOAP messages. (We use the term *specification* to denote a scheme that has been well documented but has not passed through a public standards body.) The WS-Security standard [15] from OASIS defines a framework for applying security to individual SOAP messages; GSI conforms to this standard. GSI uses these mechanisms to provide security on a per-message basis, i.e., to an individual message without any preexisting context between the send and receiver (outside sharing some set of trust roots).

WS-SecureConversation [13] is a proposed standard from IBM and Microsoft that allows for an initial exchange of message to establish a security context which can then be used to protect subsequent messages in a manner that requires less computational overhead (i.e., it allows the trade-off of initial overhead for setting up the session for lower

overhead for messages). Note that SecureConversation is only offered with GSI when using X.509 credentials as described in the subsequent section on authentication.

Both WS-Security and WS-SecureConversation are intentionally neutral to the specific types of credentials used to implement this security. GSI, as described further in the subsequent section on authentication, allows for both X.509 public key credentials and the combination of username and password for this purpose.

GSI used with either username/password or X.509 credentials uses the WS-Security standard to allow for authentication; that is a receiver can verify the identity of the communication initiator. When used with X.509 credentials GSI uses WS-Security and WS-SecureConversation to allow for the following additional protection mechanisms (which can be combined):

- Integrity protection: a receiver can verify messages were not altered in transit from the sender.
- Encryption: messages can be protected to provide confidentiality.
- Replay prevention: a receiver can verify that it has not received the same message previously.

The specific manner in which these protections are provided varies between WS-Security and WS-SecureConversation. In the case of WS-Security, the keys associated with the sender and receiver's X.509 credentials are used. In the case of WS-SecureConversation, the X.509 credentials are used to establish a session key that is used to provide the message protection.

5 Authentication and Delegation

GSI has traditionally supported authentication and delegation through the use of X.509 Certificates and public keys. As a new feature in GT4, GSI also supports authentication through plain username and passwords as a deployment option. We discuss both methods in this section.

5.1 X.509 Credentials

GSI uses X.509 end entity certificates (EECs) [17] to identify persistent entities such as users and services. X.509 EECs provide each entity with a unique identifier (i.e., a distinguished name) and a method to assert that identifier to another party through the use of an asymmetric key pair bound to the identifier by the certificate. The X.509 EECs used by GSI are conformant to the relevant standards and conventions. Grid deployments around the world have established their own certification authorities based on third party software to issue X.509 EECs for use with GSI and the Globus Toolkit.

GSI also supports delegation and single sign-on through the use of standard X.509 Proxy Certificates [17]. Proxy certificates allow bearers of X.509 EECs to delegate their privileges temporarily to another entity. For the purposes of authentication and authorization, GSI treats EECs and Proxy Certificates equivalently.

GT4 supports a delegation service which provides an interface to allow clients to delegate (and renew) X.509 proxy certificates to a service. The interface to this service is based on

the WS-Trust[14] specification (the specification is not well-defined enough to allow claim of compliance).

Authentication with X.509 Credentials can be accomplished either via TLS, in the case of transport-level security, or via signature as specified by WS-Security, in the case of message-level security.

5.2 Username and Password Authentication

GSI may use WS-Security with textual Usernames and Passwords as described in the WS-Security standard. This mechanism provides a means to support more rudimentary Web Services applications while conforming to WS-I as described in section 8.

Note that when using usernames and passwords as opposed to X.509 credentials, GSI only provides authentication and not advanced security features such as delegation, confidentiality, integrity, and replay prevention. However we do note that one can usernames and passwords with anonymous transport-level security, i.e. unauthenticated TLS as described in Section 4.1, to allow provide privacy of the password.

6 Authorization

In addition the grid-mapfile found in earlier versions of the Globus Toolkit, which provides access control based on a list of acceptable user identifiers, GT4 GSI uses the SAML standard [2] from OASIS. SAML defines formats for a number of types of security assertions and a protocol for retrieving those assertions. GSI uses SAML AuthorizationDecision assertions in two ways:

- The Community Authorization Service (CAS) [16] issues SAML AuthorizationDecision assertions as its means of communicating the rights of CAS clients to services.
- GSI uses a callout based on the SAML AuthorizationDecision protocol being defined in GGF [18] to allow the use of a third party authorization decision service, such as PERMIS[4], for access control requests to GT4-based services.

7 Future Plans

Role-based authorization is clearly an emerging direction in Grid computing. Services such as PERMIS and the Virtual Organization Membership Service (VOMS) [6] use assertions that bind attributes to users for the purposes of authorization decision making as opposed the typical identity-based authorization done today. Work has also been funded to integrate the Shibboleth service with the Globus Toolkit, providing another avenue of attribute-based authorization.

The large challenge facing attribute-based authorization today is that there is currently no standard for how attributes are communicated from the attribute authority to the relying services and no standards for expressing policy regarding those attributes.

Work is commencing on a system that leverages the ability of the WS-Security standard to allow for the transport of arbitrary credentials from a client to a service. Mechanisms in GT4 will allow for the canonicalization of those attributes and communication to an

authorization decision point as well as an authorization system to enforce simple policies based on those attributes. While the goal is that this work will be generic as to the mechanism of how attributes are expressed, the specific attribute mechanisms that will be targeted for implementation are X.509 Attribute Certificates [7] and SAML Attribute Assertions.

8 Relationship with WS-I

The goal of GSI is to adhere to the guidelines established in the WS-I Basic Security Profile 1.0 [19] where applicable. Using GSI with username and password authentication is intended to be WS-I compliant. Using GSI with X.509 credentials is intended to be WS-I compliant apart from the use of those credentials.

9 Conclusion

We have described how the GT4 Grid Security Infrastructure is an implementation of existing and emerging standards, many of which are being largely adopted by the broader Web Services community. We also describe GSI's intended compliance with the WS-I Basic Security Profile outside of its use of X.509 credentials.

For more information about Globus Toolkit security, please visit the Globus Toolkit Security web pages [11] and the Globus Technical Papers web page[12].

10 Acknowledgements

Work on GT4 GSI has been funded by DOE, NSF, and IBM. Worked described in "Future plans" is being funded by the DOE SciDAC program and the NSF NMI program.

A number of individuals have made contributions to GT4 security including: Rachana Ananthakrishnan, Karl Czajkowski, Ian Foster, Jarek Gawor, Carl Kesselman, Sam Meder, Laura Pearlman, Frank Siebenlist, Steven Tuecke, and Von Welch.

11 Document Change Log

December 8, 2004 version:

- Added description of GT4 adoption of Transport-level security throughout document.
- Added PERMIS references in Sections 6 and 7.

September 22, 2004 version: Original version

12 References

1. Simple Object Access Protocol (SOAP) 1.1, www.w3.org/TR/SOAP
2. Security Association Markup Language (SAML) Specification v.1.0, http://www.oasis-open.org/committees/security/
3. Butler, R., et al., *A National-Scale Authentication Infrastructure*. IEEE Computer, 2000. **33**(12): p. 60-66.

4. Chadwick, D.W. and A. Otenko. *The PERMIS X.509 Role Based Privilege Management Infrastructure*. in *7th ACM Symposium on Access Control Models and Technologies*. 2002.
5. Dierks, T. and C. Allen, The TLS Protocol Version 1.0, http://www.ietf.org/rfc/rfc2246.txt
6. EU DataGrid, VOMS Architecture v1.1, http://grid-auth.infn.it/docs/VOMS-v1_1.pdf
7. Farrell, S. and R. Housley, An Internet Attribute Certificate Profile for Authorization,
8. Foster, I. and C. Kesselman, *Globus: A Metacomputing Infrastructure Toolkit*. International Journal of Supercomputer Applications, 1998. **11**(2): p. 115-129.
9. Foster, I. and C. Kesselman, eds. *The Grid: Blueprint for a New Computing Infrastructure (2nd Edition)*. 2004, Morgan Kaufmann.
10. Foster, I., et al. *A Security Architecture for Computational Grids*. in *5th ACM Conference on Computer and Communications Security*. 1998.
11. Globus Project, Grid Security Infrastructure (GSI), http://www.globus.org/security/
12. Globus Project, Globus Project Technical Papers, http://www.globus.org/research/papers.html
13. IBM, et al., Web Services Secure Conversation Language (WS-SecureConversation) Version 1.0, December 18.
14. IBM, et al., Web Services Trust Language (WS-Trust),
15. IBM, Microsoft, and VeriSign, Web Services Security Language (WS-Security),
16. Pearlman, L., et al. *A Community Authorization Service for Group Collaboration*. in *IEEE 3rd International Workshop on Policies for Distributed Systems and Networks*. 2002.
17. Welch, V., et al., X.509 Proxy Certificates for Dynamic Delegation,
18. Welch, V., et al., Use of SAML for OGSA Authorization, www.globus.org/ogsa/security
19. WS-I, *Web Services Ineroperability (WS-I) Interoperability Profile 1.0a*.