Management Science and Engineering **ISSN 1913-0341**
Vol. **4**, No. **2**, **2010**, pp. 108-114 **www.cscanada.net**
www.cscanada.org

Architecture Layer Based Grid Computing Security Study[1]

BAI Qing-hai[2]

Abstract: With increasing grid computing application in more and more industries and sectors, grid security has grown into the most critical as well as important aspect in overall application architecture. This article, from architecture layer security dimension, studies security issue in grid computing environments, indicating architecture layer based security solutions in current grid computing and proposing grid security recommendations as well, which will contribute to further defining and analyzing security strategy in grid computing and function as a guideline to the development of grid computing technology.
Keywords: grid security infrastructure; security strategy; grid computing

1. INTRODUCTION

Grid concept and technology were initially introduced by Foster and Kesselman in 1998(Ian Foster, Carl Kesselman, 1998). And in 2001, Foster, Kesselman, and Tuecke defined grid as "Coordinated Resource Sharing" that resolves issues in dynamic and multi-organizational virtual structures(Foster, Kesselman, Tuecke, 2001). From then on people conduct comprehensive theoretical research and specific practice on grid computing, drawing high attention from research institutions in plenty of counties and becoming a hot research subject in IT sector.

Grid computing introduces coordinated and seamless resource sharing and computing issue. Through grids, grid computing integrates geographically scattered and systematically heterogeneous resources into a virtual "Supercomputer" (Foster, Kesselman. 2004) for largely scaled distributed and high-performance computing. Grid computing offers users huge computing capability by resource sharing and virtualization. It is right its tremendous application potential that IT makes enterprise associations hold greater expectation on grid computing as well. With its application popularization, however, researchers and IT enterprise associations start paying attention to the possible security issues from grid environments. Gird environments are based on the Internet, they therefore face security threats from various aspects, including external intrusions, internal attacks, and cross-domain call security issues, etc. Grid environment nature of heterogeneity, allocation on-demand, and distribution determines

[1] **National Nature Science Foundation of China (Approval Number: 60873235 and 60473099) and New Century Excellent Talents in University sponsored by Ministry of Education of China (Approval Number: NCET-06-0300)**
[2] Male, Ph.D Candidate, College of Computer Science and Technology, Inner Mongolia University for the nationalities**,** Tongliao, 028043, China.
* Received on February 22, 2010; accepted on May 6, 2010

that grid security covers wider area, involves more issues, and needs more complicated solutions, in addition to traditional security issues.

This article analyzes the security issue in current grid environments, and further defines, analyzes, and clarifies security solutions in grid computing environments; and it is of important significance for the development and popularization of grid computing technology.

2. ARCHITECTURE LAYER BASED GRID COMPUTING SECURITY CLASSIFICATIONS

Study grid computing security issues from architecture prospect through analyzing security issues existing in architecture layer of current grid computing environments.

2.1 Architecture level

The security issues introduced by the architecture layer involve overall grid system, including information security, strategy mapping, and decline, etc.

2.1.1 Information security

The information security in grid system is briefly classified into: secure communications, authentication, single sign-on, and agent.

2.1.1.1 Secure communications

There currently have some primary security protocols been applied in grid environments, such as SSL/SSH, SET, and S/MIME, etc.

In Globus system, SSL protocol is used to complete secure communications among GSI grid entities. In GT4, GSI supports the security in both transportation layer and session level as well[3]. To ensure transportation layer security, GSI applies SSL and X.509 credential among gird entities; to ensure session level security, GSI supports WS-Security specification and WS-SecureConversation one as well, which can provide SOAP security insurance on every session level. Since the secure work mode at session layer is to ensure the security of every single SOAP message, the work mode is able to conduct joint performance with transportation layer in any pattern and allows to impose security protections to various levels upon importance and sensitivity of data.

The reason GT4 supports session level security is that it allows users to follow WS-Interoperability's basic security profiles. GT4 services take transportation level security as default form based on user performance requirement consideration.

The lower performance implementation in session layer is primarily caused by implementation and partially by specifications. With constant integration between Web service and grid computing, grid security has started gradually steering to application session layer security. If session layer security performance gains evident improvement, GT4 will expect to use session layer security as its default mode, transportation layer security support will be ultimately turned to less important. This change will happen after transition, however, instead of happening right now.[4]

[3] http://www-unix.globus.org/toolkit/docs/4.0/security/message/
[4] The Globus Security Team.Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective. 2005.7

It uses TLS or WS security, WS security session (session level), and SOAP all together as message protection mechanism, X.509 terminal entity credential or grid user name or password as authentication credential, X.509 agent credential or WS trust as entrust, and SAML declaration as authorization.

2.1.1.2 Authentication

Credential is an important aspect in GSI authentication. In grid environment, every user and gird service must be authenticated by credential. In traditional application systems, authentication is conducted through client identity verification to ensure server security. In grid computing environments, however, in addition to the client identity verification, it also needs to conduct server identity verification, since it is possible for some imposters to provide fake services to cheat user data. In order to share information with other public key based software, GSI credential as well takes X.509 credential format.

If a user wants to apply a credential, he/she first needs to generate a pair of keys. The public key is placed in the application sent to CA authentication application, while the private one is saved in local computer in encryption pattern. If a user uses authentication credential, he/she must input password to open the encrypted file containing the private key.

CA generally includes a Register Authority that needs to verify users' credential applications, for instance, it will check if a user name is unique in CA and if it is user's true name, etc. When RA verification is completed, CA will accordingly issue the authentication application and then send the applicant a credential.

In GSI authentication, a core concept is credential. GSI identifies every user and service through credential that contains essential data to authenticate users and services.

The associations between the public keys and the entities in credentials are proved by CA. To trust user credentials and their contents, grid system then must to trust the credentials issued by CA.

GIS credential uses X.509 credential format to encrypt. X.509 credential is the most popular mechanism at present; GT4 also supports this authentication method in different levels. Although GSI X.509 credential is the proven secure solution in authentications, it must need PKI. This type of authentication mechanism has its limitations: complicated credential management institution, high maintenance costs, low operational efficiency, limited user administration scale, and requirement to keep CA online operation, etc.

Another issue is it needs to consider the interoperability with other authentication mechanisms, the interoperability with Kerberos for example. There is a method to combine GSI with kerberos, or use KX.509/KCA[5] (Kerberos-nized CA) as the gateway from GSI to Kerberos and SSL K5/PKIMT from Kerberos to GSI.

(3) Single Sign-on and agent

In traditional network application, security authentication is implemented through username/password. Various systems have their own security strategies however; users need to conduct identity authentications required by all corresponding systems before using them, users therefore have to member various usernames and passwords corresponding to different application systems, making it inconvenient for users. In addition, usernames/passwords are transmitted online and vulnerable to hacker attacks.

In view of this, GSI conducts function extension on SSL, making it have security entrust capability to reduce the number for users to input their passwords. If a grid task needs to use multiple grid resources or requests resources on users' behalf, GSI then can set an agent to avoid constant password input.

A user agent is composed of a new credential containing the identity of its owner. And the new credential is issued by user instead of CA. The credential contains both public and private keys, user

[5] http://blog.donews.com/movele/archive/2006/03/07/756298.aspx

identifier, agent identifier, and a counter indicating the time limit of the agent validity.

It is requested to ensure the security of agent private key. Due to agent time limitation however, it is not requested the absolute security that requested in user private keys. There is a processing method which is to save agent private key into a file in local machine without private key encryption. However it still needs to set file access permission for it to prevent it from being caught easily.

Once system creates and saves agent, users then can use agent credential and private key to mutually identify and password is not needed anymore. There is a slight difference in mutual identifying while user is using agent authentication. Remote resources also receive user credential in addition to agent credential. In the mutual identifying process, system uses user public key to prove the signature validity of agent credential and then uses CA public key to prove signature validity of user credential, actually creating a trust chain from CA to agent via user. Through the trust chain, the programs using the agent can run as local user programs during agent credential validity period. GSI can use both SAML and XACML to define fine-grained trust strategy descriptions.

SAML (T.Moses, ed.) (Security Assertion Markup Language) is OASIS-recommended security service standard and primarily used in secure data exchange between mutually trusted partners. It is based on XML architecture and the main protocol of federated identity authentications; it can issue a valid user identity authentication and authorization in a secure area. And standard data format supports cross-domain data exchange.

XACML(Keahey, Welch, 2002) (eXtensible Access control Markup Language) is ISO-formulated security strategy standard, provides uniform strategy description language, and improves coordinated performance capability among different organizations in WEB environment. Its owns flexibility, fits in various environments, supports data types, functions, and combinational logic, and offers high strategy expression ability, making it possible for users to define various complicated or simple rules and describe fine-grained access control needs.

Although GSI, to some extent, resolves some issues including identity authentication, single sign-on, and secure communications, etc; there are still some existing issues such as identity authentication performance, GSI identity mapping model, agent credential management, trust management service perfection, and security management costs, etc.

2.1.2 Strategy mapping issue

To implement gird computing resource access, a user must be authorized after authentication before effectively accessing resource. And grid resources can’t be reasonably applied without correct, reasonable, and flexible authorization (Keahey, Weleh, et al., 2003).

Virtual organization refers to the wide area virtual organization that gathers distributed users and resource providers with various security strategies and cross different management domains. Virtual organization aims at providing authorization mechanism and defining security authorization strategies. In addition to the special security strategies from virtual organization, there are also existing local security policies in local sites. Managing these strategies is therefore the major concern. It can classify the solutions to solve this issue into two types: resource layer issue and VO layer issue. Akenti-provided authorization mechanism is on the resource layer while CAS and VOMS-provided access control mechanism is on the VO layer.

2.1.2.1 Resource layer

After setting up a trust relationship with enterprise associations, resource provider authorizes them some access privileges. When users expect to access resources, enterprise associations issue certificate, including special policy assertion in virtual organizations. Resource provider decides to allow or decline the access request according to policy assertion and local authorized policy information.

Akenti (ThomPson, Essiari, Mudumbai, 2003;Thompson, Johnston, et al, 1999) is the product developed by Lawrence Berkeley National Laboratory, used to authorize management mechanism in GSI architecture. In Akenti authorization system, users access resources must go through resource gateway of each resource, otherwise access can be declined. Users connect resource gateway with SSL, using individual user credential for identity authentication. Resource administrator controls the access by using created credential which saved the requirements of the resource access. Clients request for accessing the resources to resource gateway, resource gateway submits authorization application to Akenti server, Akenti sever controls access through simple ACL, or through attribute credential. If it uses attribute credential, Akenti server need to access policy credential to get access requirements credential, and if there is policy information not allowed to access, the grid resource access can be declined; otherwise allow the access.

2.1.2.2 VO layer

The purpose of virtual organization is to control access towards different users belong to a specific association. The association can be granted access based on some information, such as roles, membership authentication, etc. The typical example is CAS and VOMS.

CAS (Community Authorization Server)(Park, et al, 2008) is an access mechanism developed by Globus community with purpose is to maintain access control of scalability and fine-grained in the grid environments when visiting the authorized organizations inside VO system.

Under the condition of multi-organization operations, establishment of trust relationship is difficult and complicated as well; CAS will provide uniform administration in the way of a centralized authentication service. Resource provider issues access to a global account in VO organizations and CAS server provides access control to all entities in the domain. Grid user applies access to CAS, providing user credential. After CAS server authentication, through access policy database in CAS, it decides if the access will be allowed or declined. If the access is allowed, CAS server creates a credential authorization, and the user uses it to apply for certain computing resources. A series of authentication is made by resource provider and to decide whether this access will be allowed.

Data Grid and ataTAG (Alfieri ,et al, 2005) in Europe are similar to CAS in the architecture, they both create policy credential statement from service center to grid users, then the users get access from virtual organizations. The primary difference is that they operate with different levels of authorization, CAS policy statement contains a direct user privilege which does not need the resources to further explain; in the policy statement of VOMS, it contains the list of a role or association members, grid user passes through the association relationship policy statement to resource owner, resource owner calls up the function of authorization policy decision based on local policy to decide which privilege should be granted to user.

CAS and VOMS mentioned above are based on virtual organization and give better solution and make up the drawbacks of traditional GSI. However, they have disadvantages, for instance:

①A single policy server in a virtual organization, bottleneck and reliability can be occurred during user access.

②There is no feedback mechanism in CAS, introducing negative effects during certain resource allocations.

③ Policy statement need to be further standardized. Security Assertion Markup Language (SAML) will become standardized language by international organizations.

2.1.3 Denial of service

2.1.3.1 Preventive Solution

It is divided into application filtering and intrusion detection systems. Application filtering probably occurs in XML layer; it monitors SOAP message through firewall. In application layer, new generation firewall can filter most of common DOS based on XML. There is an alternative that it uses intrusion detection system, such as IDS (Grid-based IDS system), similar to Snort system, and SANTA-G, etc.

2.1.3.2 Reactive techniques

It can be divided into link testing and logging. The fundamental principle of link testing is to find the attackers against traffic flow. However, the implementation of this technique has many difficulties, not only technical but also administrative and political, etc. Logging is to track the attacker according to data package logged in the router. However, the problem is that most of resources are requested to carry log-based identity authentication, and it is not easy to identify the attackers.

The attention on the attack of distributed denial of service has been paid more and more in recent years (Stallings, Meng, 2009). In a typical DDOS attack, a huge number of controlled hosts are used to send useless packages, preventing user from normal services. Recently, the model and tools of DDOS attack are becoming more and more complicated and efficient, making it hard to track real attackers. And the existing defense technologies are not powerful enough to resist large-scaled attacks(Chang, 2002).

During DDOS attack, an attacker initially infects a large number of hosts, so that they become zombie hosts and perform attack mission. The attack process is as following: the attacker designs the software to perform DDOS attack and hide himself; on the other hand, most of hosts have security vulnerabilities so that the attacker can install zombie software; they use scanning technique to search for vulnerable hosts. The process above has been repeated until a distributed network with a large number of zombie hosts is established.

There are briefly three ways against DDOS as following:

The early warning mechanisms before attacks, detecting and filtering mechanisms during attack, and attack source tracing and identifying mechanisms after attacks.

3. SUMMARY

The architecture layer proposes the security issues involving grid architecture. Due to the heterogeneous nature of the grid and the character that it allows virtualization in user layer, users need to consider how to map different security policies to pass through cross-domain grid, this mapping strategy is particularly important. Other security issues include attack of distributed denial of service in the grid and information security, such as data confidentiality, integrity, and authentication, etc. And all security issues mentioned are classified into this layer.

In grid computing environments, the final goal of grid security issue is to prevent grid infrastructure from known and unknown attacks. We cannot establish a once and for all grid security solution, because the attackers constantly search security vulnerabilities which include system itself, or the protocols we rely on. Therefore, the practical solution is to develop continuously, and update all solutions proposed in this article. In connection with specific security policies, whether in quality or quantity it must develop robustness security tools.

REFERENCES

Alfieri, R., Cecchini, R., Ciaschini, V., Dell'Agnello, L.,et al. (2005). From gridmap-file to VOMS: Managing authorization in a Grid environment. *Future Generation Computer Systems, 21(4)*:549-558. High-Speed Networks and Services for Data-Intensive Grids: the DataTAG Project.

Chang, R. (2002). Defending Against Flooding-Based Distributed Denial-of-Service Attacks: A Tutorial. *IEEE Communications Magazine*.

Foster, I., Kesselman, C., Tuecke, S.. (2001). The Anatomy of the Grid: Enabling Scalable Virtual organizations.*International Journal of Supercomputer Applications, 15(3).*

http://blog.donews.com/movele/archive/2006/03/07/756298.aspx

http://www-unix.globus.org/toolkit/docs/4.0/security/message/

Ian Foster, Carl Kesselman (eds). (1 November 1998).*The Grid: Blueprint for a New Computing Infrastructure (1st edition).* San Francisco, USA: Morgan Kaufmann publishers.

I. Foster, C. Kesselman. (2004). *The Grid 2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann.

Internet X.509 Public key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile, http://www.ietf.org/rfc/rfc3280.txt,2002.

K.Keahey, V. Welch.(2002). Fine-Grain Authorization for Resource Management in the Grid Environment. In Proceedings Grid Computing 2002(GRID2002). Third International Workshop, Lecture Notes in Computer Science: Security and Policy Management, 2536. Baltimore, MD, USA: Springer-Verlag.

K.Keahey, V. Weleh, S. Lang, B. Liu, S. Meder. (2003).Fine- Grain Authorization Policies in the GRID: Design and Implementation. The 1st International Workshop On Middleware for Grid Computing, 170-177.

M.ThomPson, A.Essiari, S.Mudumbai. (2003). Certificate- based Authorization Policy in a PKI Environment. *ACM Transactions on Information and System Security (TISSEC), 6(4)*:566-588.

M. Thompson， W. Johnston，S.Mudumbai，G.Hoo，K.Jackson. (1999). A. Essiari.Certifieate-based Aecess Control for Widely Distributed Resources. Proceedings of The Eighth Usenix Security SymPosium.

Park, Sang M.,Chung, Soon M. (2008). Enhanced CAS certificate for metadata-based access control in grids. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI. 2:323-329, Proceedings - 20th IEEE International Conference on Tools with Artificial Intelligence, ICTAI'08.

The Globus Security Team.Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective. 2005.7

T.Moses, ed.eXtensible Access Control Markup Language (XACML) Version 2.0,OASISstandard,http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf.

William Stallings (author), Meng Qingshu (translator). (2009). *Cryptography and Network Security –Theory and Practice* (Fourth Edition) [M]. Beijing: Publishing House of Electronic Industry, 435-437.