# RIVISIT GRID COMPUTING SECURITY

Saiful Adli Ismail[1], Zailani Mohamed Sidek[1]

[1]Faculty of Computer Science and Information Systems
University Teknologi Malaysia
81300 Skudai, Johor

Email: [1]{saifuladli, zailani}@utm.my

**Abstract**: Recent studies have shown that Grid Computing provide computational power, data storage and network bandwidth of under utilized resources at a minimal cost to the end user. It integrates geographically distributed resources and perform collaborative task. Since the goal of grid is resource sharing, computer resource will be accessed by a lot of users from different virtual organizations (VO). The security requirement becomes more vital to the Grids. Securities play a major role in providing the confidentiality of the communication, the integrity of data and the privacy of the user information. This paper focuses on to define current grid security issues and address grid security problem and challenges.

**Keywords**: Grid Computing Security, Authentication, Authorization, Confidentiality.

## 1. INTRODUCTION

Research and development efforts within the Grid community have produced protocol, services, and tools that address the challenges arising when we seek to build scalable virtual organization (VO) [1]. Current technologies for Grids security are still going on where the security is the foremost concern for the users and the stakeholder. The goal of Grid computing is to create a "virtual organization" across one or more physical organizations or "administrative domain" [2]. The most important achievement is not only secure inside the Grids but also need to secure outside the Grids.

Grid computing evolution has followed the classical path; it began in academia and is slowly moving to the enterprise community [3]. The need of holistic and comprehensive analysis of existing grid security issue and available countermeasure are very important. The ongoing Grid security infrastructure (GSI), the portion of the Globus Toolkit, can only provide fundamentally secure functionalities, but cannot prevent the stealing and masquerade of user identity, the interception and forgery of transmission, the misuse operation of resource by

privileged users and illegitimate access to sensitive data or metadata by service provider or resource owner [4]. This paper revisits grid computing security and provides an extensive literature review which focuses on current grid security issues, grid security requirement and challenges. The rest of the paper is organized as follows. Section 2 presents grid computing security overview. Section 3 describes the current status grid computing security. Section 4 discusses the latest generation of Grids. Finally Section 5 concludes the paper.

## 2. GRID SECURITY

The security of the Grid system should provide the same protection that conventional systems provide, including establishing the identity of users or services (authentication), protecting communication (encryption/decryption), determining who is allowed to perform what actions (authorization), and recording the important operations processed by the system (auditing) [4]. There are a number of basic definitions regarding computer security. Authentication is the act of ensuring that someone or something is who they claim to be. Authorization is the right to perform some action. Integrity refers to the ability of the computer system to ensure that the data is protected from unauthorized modifications. Confidentiality is the ability of computer to keep information from being disclose to unauthorized users. Nonrepudiation refers to the inability of something that performed a particular action such as a financial transaction to later deny that they were indeed responsible for the event. Trust can be defined as the assured reliance on the character, ability, strength, or truth of someone or something [5].

From [6], a Grid security standpoint, the users, the application, or the grid middleware or some combination of the three must be trusted. There are four general categories of attacks on security services such as interruption, interception, modification and fabrication. In [5] the descriptions of the attacks on the security service are: i) interruption occurs when a message is blocked to or from a particular service. ii) Interception refers to an intruder catching but not necessarily blocking a message intended for a recipient. iii) Modification refers to the action of interpreting, modifying, and then retransmitting a message to a security service and iv) Fabrication refers to generating a new message from scratch and attempting to insert it into the normal message flow.

Grid computing security can be convincing and widely use by the users such as scientists and engineering in any field or discipline when security is the foremost concern for management, implementer and developer. Such of security is difficult to find in the environment of Grids.

From [3] had categorizes grid security issues by classified as Host, Architecture and Credential Level.

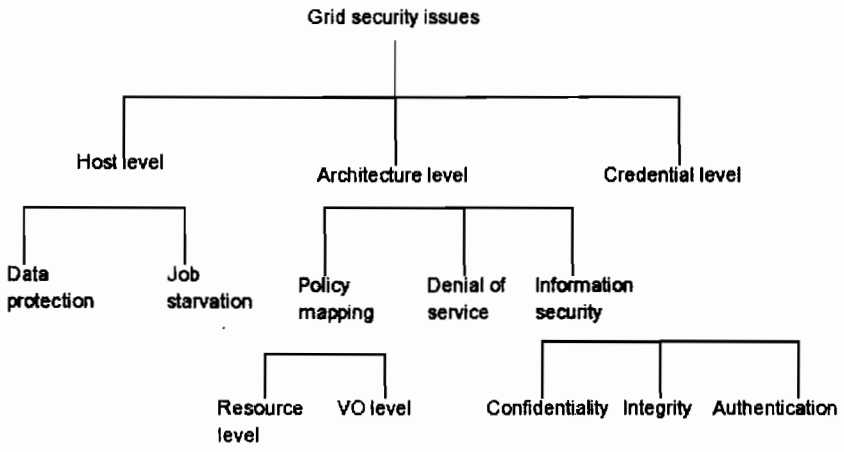


Figure 1. Taxonomy of Grid Computing security issues.

First, let's look at the term use in figure 1, where Host refers to a computer system that includes PCs and servers. We can define a grid resource as a computing or data element within the grid. The main difference between a host and a resource is that a host can affiliate itself into the grid and become a computing resource. Finally, credentials are tickets or tokens used to identify, authorize and authenticate a user.

Host level issues correspond to an on-demand grid system where the host affiliating itself into the grid system. The main sub issues here are data protection and job starvation. Architecture level issues address the concern of the grid system as a whole. Issues like information security, authorization and service level security generally destabilize the whole systems and hence an architecture level solution is needed to prevent those. Credential level issues become very important in the grid context as there are multiple different systems which required varied credential to access them [7]

## 3. ISSUES IN GRID SECURITY

According to [4] the security of the grid may affect many area of your system, from who is allowed to access the grid to which machines in your grid are able to perform different operations. Security issues in a Grid can be categorized into:

I. System level Security: It deals with the problem of running a foreign application such as of viruses, worms, malicious codes etc.

II. Architecture level Security: It deals with the development of secure infrastructure for a Grid system, which includes authentication, authorization and confidentiality, in Grid environment.

III. Interoperability: It deals with managing a heterogeneous security infrastructure in terms of different security measures like authentication, authorization etc. among multiple domains in an organization or larger network.

After reviewing the security issues in Grid Computing environment, the categorization of the grid security issues are also nearly equal that has mention in [3].

### a) Grid Security Requirements

Security requirement within the Grid environment are driven by the need to support scalable, dynamic, distributed virtual organization (VOs) where the collection of diverse and distributed resources that seek to share and used the resource in a centralize coordination fashion. For better understanding for security requirements for grid environment, [8] give some example that concerns a scientific experiment called Compact Muon solenoid. In the experiment, conducted at the Large Hadron Collider in the CERN Laboratory, Switzerland, the collected data is to be analyzed by more than 2000 physicist at more than 150 universities and laboratories located in 34 countries. Ideally, the security challenges mainly come from dissemination, processing and sharing of the data.

The process of dissemination of the data from one resource to another, need the authenticity of the requestor be verifiable so that only authorized requestor is allowed to access the available resources. Therefore, dealing with dissemination and access of data across many different countries, integration of security mechanism and policies becomes a requirement. In many cases, data confidentiality and integrity can be vital to safeguard the scientist research finding. When it comes to processing data, high end processing resources generally required high investment and thus it is desirable that their usage can be tightly controlled, possibly through access control mechanisms. This is needed in balancing the resource usage between

users from the physical organization and the remote user such as physicist from virtual organization. From data sharing aspect, trust establishment between entities of various universities and laboratories plays a crucial part in grid security. Policy enforcement can be more complicated because of the exchanging policies among the various virtual organizations that have differences security mechanisms and access privileges.

With a lot of papers and journal say about the requirement of the grid security, [4, 8] has compile a more formal list of grid security requirement that is necessary for supporting scalable, dynamic and distributed virtual organization such as authentication, authorization, confidentiality, single sign-on, delegation, non-repudiation, accounting, auditing and integrity. In this paper we will cover only three of them, where it was relevant to our research. The first security requirement is authentication, where in grid environment there are several types of entities that need to be authenticated. The most common are individual users who utilize grid resources and host which provide resources and services. Secondly, is authorization where grid application required access to resources which may be located in different organization domains with different owners and thirdly, is confidentiality where as with other standard distributed systems, protection of sensitive information from exposure to unintended parties can be critical.

**b) Authentication**

Users are normally authenticated by a resource as a step that allows for establishing their authorization privileges, as well as for auditing and account. Authentication deals with one party gaining the assurance that the identity of another is declared and true, preventing impersonation. Entity authentication is the process of one party in the communication process being assured through quantitative evidence that the identity of a second party in the communication process is correct and allowed [9]. In [4] has mention, that there are many methods available for user or system authentication. Three prominent authentication models are public key infrastructure (PKI), Kerberos and secure shell. The detail discussion regarding this model will discuss below:

1. Public Key Infrastructure (PKI) – A public Key Infrastructure comprises of two keys. The first one is the Private Key and the other is public key. A sender encrypts his message with his private key and sends it to receiver. The second key is public key where is given to the receiver to decrypt the encrypted message by the sender. To

validate the keys, a reliable body is chosen who is responsible for certifying the keys used for communication. This body is known as Certification Authority, who in turn is registered with a Registration Authority (RA). This whole process of validating the CA by RA and further user keys by CA come under PKI.

2. Kerberos – Kerberos [4]is a distributed authentication service that allows a process (a client) running on behalf of a principal (a user) to prove its identity to a verifier (an application server, or just server) without sending data across the network that might allow an attacker or the verifier to subsequently impersonate the principal. Kerberos optionally provides integrity and confidentiality for data sent between the client and server.
3. Secure Shell – it is currently one of the most widely used secure communications protocol on the internet. Secure Shell [4] provides three main capabilities, which opens the door for many creative secure solutions:
    a. Secure command shell: it enables remote login and provides a user with a facility to authenticate and switch off a user in between who is trying to maliciously harm the system.
    b. Secure file transfer: it make the use of secure file transfer protocol to securely transfer the files.
    c. Port forwarding: it provides a user with the facility of data tunneling by securing the information passed.

**c) Authorization**

Authorization in a computer network is the permission for a user or system to access a particular object (computer, resource, data, or program). Authorized user should, in practice, be the only ones allowed to access specific data or resources [9]. In the authorization credential mode, [10] the user has identity credential of their own (issued by either the community, the resource owner, or third party) and the science gateway augments these by providing authorization credential that the supplied to the resource along with the user's identity credential. According to [5], there are two general approaches for authorization: identity-based or token-based. Identity-based approaches are typically associated with access control lists, while token-based approaches are also referred to as capability-based authorization. In identity-based approaches, only the authenticated user identity (and the requested action) is presented to the resource, which then checks an internal list of allowed identity/action pairs. In token-based approaches, an unforgeable token is granted to the user, who the presents it to the service as proof of her right. In some sense, the service does not

care who the presenter is, rather just that the request came with the appropriate token. A drawback of identity-based approaches is that identity-based approaches cannot easily support delegation, in which one person allow/requests another user or software agent to act on the user's behalf. On the other hand, a drawback of token-based approach is that it may be very difficult to dynamically revoke access rights.

In [4] has mention that, there are several methods to implement authorization such as by having the following methods:

I. Access Control List (ACL) – Access control list is the mechanism that allows owners of resources to define manage and enforce access conditions applicable to each resource. Access control deals with the configuration of users and those actions that they should be allowed to do. Within the database, access control involves creation of users and granting them the roles and privileges to do what they need to do to accomplish their jobs [9]. When it comes to the Grid, this scheme is not compactable, because each virtual organization (VO) and traditional organizations will have their own policies for access control. These policies and the access control based on them are high dynamic and heterogeneous or even conflicting. And it is not reasonable to expect that heterogeneous systems for different purposes and under control of different parties will be able to define a common homogeneous set of access control and authorization criteria, object or a resource.

II. Role Based Access Control (RBAC) – Role-based access control is being increasingly recognized as an efficient access control mechanism that facilitates security administration. Roles are identified with various job functions in an organization and users are assigned to roles based on their job responsibilities and qualification. Permissions through the roles allocated for them. This feature of role-based models greatly simplifies the management of permissions [11]. RBAC in [4] has mention that the development of RBAC coincides with the advent of corporate intranet. The roles are collections of entities and access rights grouped together based on different tasks they perform in the system environment.

    a. If a user moves to a new function within the organization, the user can simply be assigned to the new role and removed from the old one
    b. Whereas, in the absence of an RBAC model, the user's old permissions would have to be individually revoked, and new permissions would have to be granted

III. Community Authorization Service (CAS) – CAS [7] has been developed by the Globus Toolkit. CAS look at the problem of scalable representation and enforcement of access policies within distributed virtual communities. The problem of authorization is handled using a trusted third party called the Community Authorization Service server which is

responsible for managing the policies and governing access to the community's resources.

IV. Virtual Organization Membership Service (VOMS) – VOMS is the authorization system developed for the European Data Grid (EDG) as part of the DataGrid and DataTag projects [7]. The server is essentially a front-end to an RDBMS, where all the information about users is kept. The VOMS system is composed by the following parts [12]:

   a. User Server: receives requests from a client and return information about the user.
   b. User Client: contacts the server presenting a user's certificate and obtains a list of group, role, and capabilities of the user.
   c. Administration Client: used by the VO administrator (adding users, creating new groups, roles, etc)
   d. Administration Server: accepts the requests from the clients and update the Database.

**d) Confidentiality**

Issues with confidentiality are more complex in that the requirements are less easily defined. Most of the issues in this area are concerned with what it is acceptable for a given individual to see [13].

a) Communication – Transmissions between parties should be secured to allow the transport of sensitive or private data and / or programs. This becomes more important with the rising role of VO's within the grid.
b) Data Protection – In order for grids to legally exist it is essential that they comply with appropriate legislation, such as Data Protection Act.
c) Multi-level services – In the case of multi-level services agreed information flows must be set out before a service is utilized, to take into account what information if any should be passed on of a user's identity and / or information about them to subsequent sub services.
d) Secure Areas – Users should have a secure area from which to run programs. Data should not be accessible to others from outside nor should the program be able to affect other processes running on a machine outside of that area. (sandbox approach)

From [4] we have two popular models for confidentiality. When the keys are to be passed between sender and receiver public key cryptography model is used and for communication or message passing secret key cryptography model is used.

a) Public Key Cryptography – public key cryptography primarily focuses on solving issues related to key sharing and key distribution that occur in symmetric encryptions. The main purpose of public key cryptography is to allow distance users to communicate securely over an insecure channel without worrying about the "agreed upon" key before communication is established. Public key cryptography contains two related keys, the private key (to be kept secret) and the public key (for distribution). Public key cryptography also called asymmetric-key cryptography simply because not all the parties have same information during the identity recognition and data transformation process. Digital signature is one of the applications using public key cryptography [10].
b) Secret Key Cryptography – It makes use of symmetric encryption [4]
   i. Both parties use the same key value to encrypt clear text into cipher and to decrypt a cipher text back into clear text.
   ii. Symmetric key is how to exchange the secret key between parties who do not trust/know each other such as on the Internet.
   iii. For a group of n users, the number of keys to be managed is $K = n(n-1)/2$

## 3. CURRENCT STATUS OF GRID SECURITY

In Grid security there is various security challenges faced such as protecting applications and data from system where computation executes, stronger authentication needed (for users and code), protect local execution from remote systems [4]. The current Grid security standard and popular use in the grid environment are as follow:

### 3.1 Grid Security Infrastructure (GSI)

Part of the Open Grid Service Architecture (OGSA) function, GSI provides authentication and secure communication over open network connections. The GSI implementation adheres to the Generic Security Service Application Programming Interface (GSS-API), which is a standard development tool for security system promoted by the Internet Engineering Task Force (IETF) [10]. In [4] has mention that GSI deals with interdomain operations, bridging the different local security solution of constituent sites.

- Credential, using standard X.509v3 certificates as the private keys, represents the identity of each entity such as user, resource, program specifying the entity's name and additional information, such as a public key. A certificate authority (CA) is a trusted third party ties an identity to a public private key pair signing a certificate.
- An authentication algorithm, defined by the secure socket layer version 3 (SSLv3) protocol, checks the entity's identity. The veracity of entity's identity is only as good as the trust placed in the CA that issued the certificate, so the local administrator installs these certificates, which are then used to verify the certificate chains.
- An entity can delegate a subset of its right—such as a process a program creates—to a third party by creating a temporary identity called a proxy. Proxy certificates can form a chain, beginning with the CA and growing, as first the user, then the user's proxies, signs certificates. By checking the certificate chain, process started on separate sites by the same user can authenticate to one another by tracking back along the certificate chain to find the original user certificate.
- Each resource can specify its policy for determining whether to accept incoming requests. The initial GSI used a simple access control list, but the current version uses other techniques.
- The authentication protocol verifies the global identity of involve parties, but GSI must convert this name to a local subject name such as a login name or Kerberos principal before the local security system can use the name. GSI does this by consulting a simple text-based map file under the local site's control that defines the binding between global and local names.
- The standard interface GSS-API provides access to security operations. GSI uses OpenSSL or SSLeay, the free implementation of SSLv3, for its authentication protocols and support for proxy certificates. SSLv3 is used widely for web security, has been well scrutinized for security problems, and has broad acceptance as a mature protocol.

### 3.2 Open Grid Security Architecture (OGSA)

The Open Grid Security Architecture (OGSA) introduces both new opportunities and new challenges for grid security. It makes sense that OGSA security should leverage as much as possible the existing and emerging web services security specifications address standard methods for authentication and establishment of security contexts and trust relationships in WS-SecureCoversation and WS-Trust standard formats for security token exchange in WS-

Security and SAML; and expression of web service security policy in WS-Policy and XACML [8]. These specifications have been exploited by grid developers to create uniform and inter-operable methods to be used in grid security. For more details and examples of OGSA security and web services, see [14].

In [4] has mention that the OGSA roadmap proposed leverages other WS security specification as much as possible, when situations where other specifications are still no be available at the time they are needed, tactical solution are proposed that should be reconciled with the appropriate specification when they become available.

### 3.3 WS-Security

The WS-Security specification applies to all simple object access protocol (SOAP) extensions to secure message security (using different mechanisms such as XML signature and XML encryption) [16]. XML Signature [8] provides an XML syntax defined for digital signatures. An XML signature can be applied to some or all the content of one or more XML documents or SOAP messages.

According to [4] WS-Security also provides a general-purpose mechanism for associating security tokens with messages. However, no specific type of security token is required by WS-Security. It is designed to be extensible (e.g. support multiple security token formats) to accommodate a variety of authentication and authorization mechanisms. For example, a requestor might provide proof of identity and a signed claim that they have a particular business certification. A web service, receiving such a message could then determine what kind of trust they place in the claim.

Table 1. WS-Security Functional Categories [16].

| WS-Security Standard | Functions |
|---|---|
| WS-Policy | Defines how to express capabilities and constraints of security policy |
| WS-Trust | Describes the model for establishing both direct and brokered trust relationships (include intermediaries) |
| WS-Privacy | Enables users to state privacy preferences and Web services to state and implement privacy practice. |
| WS-SecureConvesation | Describe how to manage and authenticate message exchanges between parties, including exchanges of security contexts, establishing and deriving session keys. |
| WS-Federation | Describe how to manage and broker the trust relationship in a heterogeneous federated environment, including the support of federated identities |
| WS-Authorization | Defines how Web Service manage authorization data and policies. |

From Table 1, given that there are a total six Web service specific security feature, how do we know their relationship to one another? To answer the question, [16] has presents the six WS-Security functional categories as a triangle composed of the distributed message-based security components of trust, interoperability and integration as illustrated in Figure 2.

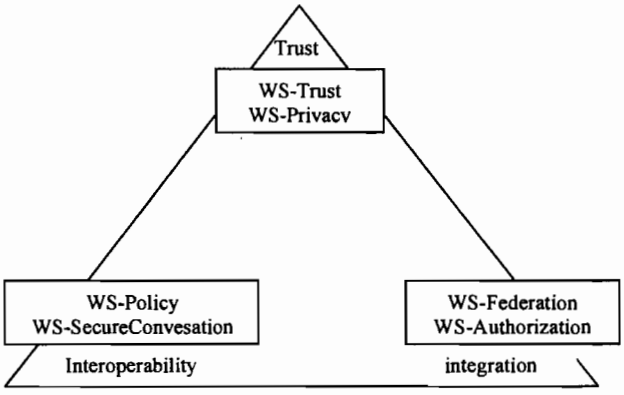


Figure 2. WS-* Security Triangle [16].

### 3.4 Security Assertion Markup Language (SAML)

SAML is an XML standard for exchanging authentication and authorization data between security domains, that is, between an identity provider (a producer of assertions) and service provider (a consumer of assertions) [4]. SAML tries to address three scenarios – Single Sign-On (SSO), authorization, and back office transactions. SAML uses Application Programming Interface (API) within Web Services to obtain trust services in order to obtain authorization and authentication assertions about the individuals and entities [16].

The main purpose of SAML is to allow trust assertions to be specified in XML and to focus on the authentication and authorization aspects of security. According to [16], SAML is fundamentally a combination of three XML-based mechanisms:

i. Assertions – SAML is an XML schema and definition for security.
ii. Protocol – SAML is an XML schema and definition for a request/response protocol.
iii. Binding – SAML rules are described as sets of binding and protocols.

### 3.5 Extensible Access Control Markup Language (XACML)

Extensible Access Control Markup Language (XACML) [15] provides a policy language which allows administrators to define the access control requirements for the enterprise resources. The language and schema support include data types, functions, and combining logic which allow complex (or simple) rules to be defined. XACML also includes an access decision language used to represent the runtime request for a resource. When a policy is located which protects a resource, functions compare attributes in the request against attributes contained in the policy rules ultimately yielding a permit or deny decision.

When a client make a resource request upon a server, the entity charged with access control by enforcing authorization is called the policy enforcement point. In order to enforce policy, this entity will formalize attributes describing the requester at the policy information point and delegate the authorization decision to the policy decision point. Applicable policies are located in a policy store and evaluated at the policy decision point, which then returns the authorization decision. Using this information, the policy enforcement point can deliver the appropriate response to the client.

## 4. LATEST GENERATION GRIDS

In this section we will look at some of the technologies that may have an impact on the grid security landscape in the future. According to [4] the current focus in Grid security is developing protocols that will prevent unauthorized access while enabling interoperability between diverse Grid sites. There are three security challenges identified by Global Grid Forum (GGF) for Grid environment:

### 4.1 Integration

Integration among the existing grid in the world is impossible to done, if the standard of grid middleware, protocol and the policy is different between each others. It is unrealistic to use a single security technology to address Grid security issues. Grid security architecture must be:

i. Implementation agnostic: It can be instantiated in term of any existing security services. A good example is authorization. We have seen a rapid evolution in the past years of different authorization technologies [17] (grid-map files used by globus, VOMS, CAS, PERMIS) that are all non-interoperable. In addition, different deployment projects have hitherto adopted or supported a single version of these technologies.
ii. Extensible: It can incorporate new security services as they become available
iii. Integrate: Existing security services.

### 4.2 Interoperability

It is important that the security architecture used by any Grid environment allows for basic interoperability with other Grid deployment or middleware project such as the Open Science Grid [18], OMII [19], NorduGrid [20] and LCG [21]. At a minimum, it should be possible to use the authentication and authorization architecture with system that is already in operation.

In [4] has mention, services that traverse multiple domains and hosting environments need to be able to interact with each other, thus introducing the need of interoperability at multiple levels. The security of Grid project must be able to interoperate with the local security solution at different levels:

i. At the protocol level, mechanisms that allow domains to exchange messages are required.

ii. At the policy level, each entity can specify its policy and the policy can be mutually comprehensive.

iii. At the identity level, mechanisms to identify a user from one domain in another domain are required.

### 4.3 Trust Management

In a controlled grid environment, trust can be managed using static configuration. However, with the growth of flexible grid system, the need for managing trust becomes more and more important. Trust management is the activity of collecting, encoding, analyzing and presenting evident relating to competence, honesty, security, or dependability with the purpose of making assessments and decision regarding trust relationships [4].

Trust management systems (TMS) responsible for managing trust in a distributed environment. TMS system can be divided into two main types: policy-based TMS and reputation-based TMS [7]. A Grid security service request can span multiple security domains. The security domains involved to meet a Grid service request required establishing trust each other. Due to the dynamic nature of a grid environment, it is unfeasible to establish end-to-end trust prior to execution of an application. The issues of trust establishment become complicated with transient Grid services [4].

## 5. CONCLUSION AND FUTURE WORK

This paper presents the Grid Computing Security by revisit the previous and latest research in grid computing security. The purpose to revisit grid security is to identify areas of grid computing security which more extensive research is needed. More important, this paper contributed to the overall body of knowledge and research concerning security in Grid computing. Standard procedures for developing Grid applications and frameworks must include security as part the larger design process, particularly because the Grid presents unique security challenges. Therefore, our future work we will consider to looking at Identity-Based Encryption in grid computing environment especially to achieve confidentiality which is a new cryptographic technique where encryption can be done using any known string associated with the receiver.

**REFERENCES**


[1] Mukhin, V., "The Security Mechanisms for Grid Computers," IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Application, 6-8 September 2007.

[2] Welch, V., Siebenlist, F., Foster, I., Bresnahan, J., Czajkowski, K., Gawor, J., et al. (2003). Security for Grid services. In *High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on* (pp. 48-57).

[3] Chakrabarti, A., Damodaran, A., and Sengupta, S., "Grid Computing Security: A Taxonomy," IEEE Security and Privacy, 2007.

[4] Bhanwar, S., and Bawa, S. "Securing a Grid," PWASET, Volume 32, August 2008.

[5] Humphrey, M., Thomson, M. R, and Jackson, K. R., "Security for Grids," Proceeding of the IEEE, Volume 93, Issue 3, 2005.

[6] Butt, A. R., Adabala, S., Kapadia, N. H., Figueiredo R. J., and Fortes, J. A. B., "Grid-computing portals and security issues," Journal of Parallel and Distributed Compuitng, Elsevier, 2003.

[7] Chakrabarti, A., "Grid Authorization System," In Grid Computing Security. Chapter 5. Springer-Verlag, 2007.

[8] Lim, H. W., "On the Application of Identity-Based Cryptography in Grid Security," Doctoral Thesis, Information Security Group, Department of Mathematics, Royal Holloway, University of London, (2006).

[9] Wells, A. J., "Grid Application System Design", Chapter 4, New York, Taylor & Francis Group, 2008.

[10] Welch, V., Barlow, J., Basney, J., Marcusiu, D and Wilkins-Diehr, N., "A AAAA Model to Support Science Gateways With Community Accounts ," Concurrency and Computation: Practice and Experience, Willey InterScience, 2006.

[11] Crampton, J. and Khambhammettu, H., "Delegation in role-based access control," International Journal, Information Security, Springer, 2008.

[12] Alfieri, R., Cecchini, R., dell'Agnello, L., Frohner, A., Gianoli, A., Lorenty, K., and Spataro, F., "VOMS, an Authorization System for Virtual Organizations", SPIE International Conference on Performance and Control of Network Systems, November 1997.

[13] Lock, R. and Sommerville, I., "Grid Security and its use of X.509 Certificates," IEEE-ACM Transactions on Networking, 3(3), June 1995.

[14] Siebenlist, F., Nagaratnam, N., Welch, V., and Newman, C., "Security for virtual organizations – federating trust and policy domains," In Foster, I., and Kesselman, C.,

editors, the Grid: Blueprint for New Computing Infrastructure, Chapter 21, San Francisco, Elsevier, 2004.

[15] Moses, T., "eXtensible Access Control Markup Language (XACML) Version 2.0", OASIS Standard, 2005.

[16] Chiang, B., "A Feasibility Study:Secure Public Key Infrastructure with Quantum Key Distributions in Grid Computing", Doctor of Professional Studies in Compuing, The Ivan G. Seidenberg School of Computer Science and Information System, PACE University, January (2007).

[17] EGEE Security JRA3, "Global Security Architecture For Web and Legacy Service", 2004.

[18] Open Science Grid (OSG). http://www.opensciencegrid.org/.

[19] Open Middleware Infrastructure Initiative (OMII). http://www.omii.ac.uk/.

[20] Nordugrid. http://www.nordugrid.org/.

[21] LHG Computing Grid Project (LCG). http://lcg.web.cern.ch/LCG/.