Towards Novel And Efficient Security Architecture For Role-Based Access Control In Grid Computing

M.Nithya
PhD Full Time Scholar
Dept of Electronics and Communication Engineering
Government College of Engineering
Salem, Tamil Nadu, India

R.S.D.Wahida Banu
Professor/Head
Dept of Electronics and Communication Engineering
Government College of Engineering
Salem, Tamil Nadu, India

Summary
Recently, there arose a necessity to distribute computing applications frequently across grids. Ever more these applications depend on services like data transfer or data portal services and submission of jobs. Owing to the fact that the distribution of services and resources in wide-area networks are heterogeneous, dynamic, and multi-domain, security is of vital significance in grid computing. Authorization and access control; the significant aspects of security, have attracted increased attention in grid computing. Role Based Access Control (RBAC) is an emerging access control mechanism in grid computing. RBAC was afforded in the Globus toolkit with the support of Community Authorization Service (CAS) and this CAS was employed by several researchers in providing access control. The major problem with the CAS is that the user credentials are revealed to the virtual organization (VO) thereby leaving them in jeopardy. Moreover, once the user credentials are hacked, both the user and VO resources become vulnerable. In this paper, we have proposed a novel architecture for Role Based Access Control in Grid computing where user credential and security are regarded as a prime concerns while sharing data and computational resources in a grid problem. The evaluation mechanism detailed in this paper is highly resistant for both the users as well as for the VO resources. In the proposed mechanism, the user credentials are not revealed to the VOs, thus protecting the users from hacking possibilities. Since the hacking possibilities of user credentials are reduced the proposed system also prevents VO resources being hacked by some adversary users of the organization. This makes our model more efficient when compared to other models.

Key Words: *Grid computing, Grid security, Authorization and Access Control, Role Based Access Control (RBAC), Community Authorization Server (CAS), Virtual Organization (VO), User credentials.*

1. Introduction

Grid computing is considered as a budding technology of enormous potential in the industry as well as in academia [1]. Resource sharing with scalability and heterogeneity is facilitated by a grid environment [2], [3]. Security gains chief significance while distributing data and computational resources in a grid. Being a computing environment which facilitates resource sharing with scalability and heterogeneity, the security system in grids engrosses authentication and authorization [4], [5]. The problem of grid authentication has been under research for quite sometime now. The authorization based on the collective security policy from the resource provider and virtual organization (VO) and the authentication of the user are essential for security [6]. In grid computing, the concept of virtual organization can be put forward as a set of participants with various relationships that intend to perform a task by sharing resources. Data, computers, scientific instruments, software and more comprise the resources [2]. The highly dynamic availability of users and resources in VOs make it difficult to predict them. Moreover, it is necessary to define and enforce some sort of VO-wide access control, usage and related management policies [7].

The number of users and applications inside a majority of the grid environment and virtual organizations is on a steady rise. This has resulted in the requisite of scalable solution to administer authorized access. One of the vital components of security services is the access control [8]. The sensitive information and resources in information systems are shielded from illegitimate access using the access control that is broadly employed as a security mechanism. Only the privileged entities with a business requiring to access are permitted by the access control. Several models that have been recently presented in relation to access control which include Role Based Access Control (RBAC), Discretionary Access Control (DAC), Mandatory Access Control (MAC), Context Based Access Control, most recently the RBAC model [9]. The researchers were inspired to think of ways in which RBAC could be integrated into grid environment owing to its

Manuscript received March 5, 2009
Manuscript revised March 20, 2009

evolution as a reliable standard for single enterprises. In comparison to other access control mechanisms, RBAC has several advantages.

RBAC permits permissions to be handled in terms of user job roles, thus simplifying the access control administration and presenting better manageability in enterprise environments [10]. Any user, even the security administrator does not have the discretionary rights to allocate access rights. Rather than updating privileges for every user on an individual basis, barely the access privileges of the users essential to perform their duties or role and updates that can be done to roles are given to the users [11]. Database management systems, operating systems and middleware architectures extensively use RBAC to offer access control. Access rights (permissions) in RBAC are associated with roles and a set of roles in RBAC include the users as members. A user gains all the permissions of a role in the system, when it is a member of that role. Thus the permission to model the security infrastructure of that organization in proportion to its business use cases, to assign a role to a set of use cases and to assign users to the roles associated with the use cases they need to perform is granted to the organization [12], [13].

In case of grid computing, the Virtual Organizations (VOs) [2] proffer the coordination essential to make the geographically-diverse heterogeneous resources possessed by a multitude of institutions accessible to the members of the organization. The members of the virtual organization are the proprietors of these resources and the members desire to sustain a local control over their resources. This has necessitated the need for access control mechanisms in virtual organizations. A range of requirements for the grid security in VOs have resulted in the recognition of RBAC as a potential mechanism among the wide variety of access control mechanisms available in the literature. Users require globally defined names that are acknowledged at all sites they access. It is necessary for the user's identity to pass securely and transparently amid sites while the jobs progress [4]. It must be possible for the users to access resources in a dynamic fashion devoid of any administrator intervention. These resources need to be synchronized appropriately and must interrelate securely with other services. Therefore it is essential for the resources to possess global identities and they need to be accessed without any local policy violation [15].

RBAC displays apparent advantages over conventional discretionary and mandatory access control models in such environments since it facilitates the uniform representation of diverse security policies and guarantees that no security violations take place during inter-domain access [16]. Additionally, RBAC is renowned by its innate support for the Principle of Least Privilege [17]. A number of researchers have employed RBAC for offering controls in grid computing [13, 10, 15, 18-22]. The contemporary standard in effect for security in grid computing is the security component constituted in the Globus toolkit [14]. RBAC was afforded in the Globus toolkit with the support of Community Authorization Service (CAS). The CAS records user groups and their corresponding permissions on resources besides targeting access control for resources.

Of late, the Community Authorization Service (CAS) offered by the Globus Toolkit was used to support the RBAC within the Open Grid Services Architecture-Data Access and Integration (OGSA-DAI) framework by Anil L. Pereira et al. [15]. In their approach, a proxy credential which is signed by user's own credential is created by the user. The CAS server is offered with the proxy credential, which returns the CAS proxy credential, a novel credential in which the CAS policy assertions representing the user's capabilities and restrictions in the form of extension are included. The resource provider is offered the CAS proxy credential. The validity of the proxy credential is established by the resource provider who then obtains the restrictions imposed by the CAS server by parsing the CAS policy assertions, based on which the resource provider will provide access. Owing to the fact that all CAS credential contains information that identifies the user, their method does not provide privacy protection for the users, even though they provide security in terms of access control. In addition, the adversary user will be able to effortlessly exploit the resources belonging to the provider by hacking the user's credentials in that resource provider.

In this paper, we have proposed a novel architecture for grid computing role based access controlled VOs sharing their resources where security is considered as an issue for both the organizations as well as the users of different organizations. The concept of securing and providing access to the shared resources for the authorized users as well as securing the user credentials so that it is not revealed to the VO is discussed in this paper. In our model, we have implemented an analysis based on evaluating the keys that are generated by the user, CAS and VO. The key that is generated by the user is changed for every access it requests to the VO. The keys for CAS as well as VO are generated and maintained periodically depending on the nature of the organization. This concept is implemented and checked successfully.

The design flow of this paper is as follows: A brief review of the researches related to RBAC in grid computing is detailed in Section 2. The proposed architecture and the steps are given in Section 3. In Section 4 the proof for our implemented architecture is being discussed. The experimental results are provided in Section 5, and finally we conclude in Section 6.

2. Review Of Related Works

Several prior works correlated with providing security in grid computing employing role based access control are the motivation behind or work. The following section provides a brief assessment of some of the works:

A RBAC method for Grid database services in Open Grid Services Architecture-Data Access and Integration (OGSA-DAI) was projected by Anil L. Pereira et al. [15]. In case of OGSA-DAI, access control results in considerable administration overhead for resource providers in virtual organizations (VOs) since each of them have to deal with a role-map file comprising of authorization information for individual Grid users. In order to solve the aforesaid problem, they employed the Community Authorization Service (CAS) proffered by the Globus Toolkit to assist the RBAC within the OGSA-DAI framework. The access control technique presented enhanced the manageability for a huge number of users and condensed day-to-day administration tasks of the resource providers besides the fact that they uphold the ultimate authority over their resources. Assessment of the performance revealed that the method includes negligible overhead to the existing security infrastructure of OGSA-DAI.

Architecture for the integration of authentication and authorization schemes for constructing a secure Grid system was presented by Jongil Jeong et al. [18]. SAML (Security Assertion Markup Language) and XACML (eXtensible Access Control Markup Language) play significant solution roles in incorporating single sign-on and authorization in their method. However, they suggested SAML as a substitute to the existing standard that suggested by IBM and Microsoft. Therefore the architecture paved way for the possibility of implementing a variety of single sign-on technologies in building secure Grid computing. Furthermore, they as well suggested XACML that provides Grid computing with an efficient way to implement role-based access control.

An advanced model for RBAC policies was presented by Benjamin Aziz et al [13]. They also defined a risk measure for the model, which conveys elements of the operational, combinatorial and conflict of interest risks existing in a particular policy example. The model comprises of risk-reducing mechanisms analogous to practical mechanisms such as firewalls, stack checking, redundancy, and event tracking which are commonly employed to bring down risks in real systems. Besides, they as well described policy transformation operators that generate fresh policies that permit the behaviors of the old policy besides potentially decreasing the risk measure.

The issues concerned with the design and rapid operation of large scale secure information sharing (SIS) systems for coordination involved with multiple agencies was explored by Ganesh Godavari et al [10]. Procedures and tools were built in order to swiftly set up the public key infrastructure (PKI) and privilege management infrastructure (PMI) for the multi-agency SIS systems. A multi-agency SIS testbed works on basis of the LDAP servers and web servers were created to discover the utilization of the attribute certificate, public key digital certificate, and role-based access control for safe access and efficient authorization. The key contribution of their work was the building of framework that employs PKI, RBAC, PMI, and web-services for information sharing on basis of authentication, authorization, and access.

Lorenzo Cirio et al [19] demonstrated the manner in which Semantic Web technologies can be employed to create an access control system. They pursued the RBAC approach and enhanced it with contextual attributes. The methodology provides for the dynamic association of roles with users. Classification of both users and resources and the verification of the consistency of the access control policies were carried out with the aid of a Description Logic (DL) reasoner. They reduced the restricted expressive power of the DL formalism by cleansing the output of the DL reasoner with SPARQL queries.

The trust management and RBAC were combined by Chen Ying et al [20] to build a dynamic-role based access control framework. The framework was found to successfully improve the function of access control in grid environment, and provide appropriate punishing methods to vicious entities. Furthermore, it abridges the management, proffers probabilities to find trust relations amongst entities and enhances the scalability of the system.

The resemblances between trust management and distributed access control systems were illustrated by Nathan Dimmock et al. [21] through a demonstration on how the OASIS access control system and its role-based policy language can be enhanced to arrive at decisions on the basis of trust and risk assessment other than solely on the basis of credentials. They applied the model to the prototypical instance of a file storage and publication service for the Grid, and assessed it with the aid of the Prolog based OASIS implementation.

An architectural framework for adaptation and implementation of RBAC for grid access control was proposed by Geethakumari et al [22]. The methodology comprises of solutions for allocation and revocation in a single domain grid enterprise. The conventional Role Based Access Control, despite being an effective access control standard, does not deal with the issue of resolving a local role into a global role. Hence, they as well provided

an architecture on basis of RBAC that is capable of establishing role equivalence among the domains by mapping a local domain role to its corresponding global role. The ultimate authorization decision was taken on basis of the mapped global role ranking and the resource access policies as well.

3. Novel And Efficient Security Architecture For Role-Based Access Control In Virtual Organizations

Recently, many researches are conducted on the basis of improving the security of shared resources between the organizations in a VO, but the security of the users is being left under a threat since the organizations in a VO are changing dynamically. For joining a VO an organization should possess the identity certificate for the corresponding VOs. Access to the VO resources is provided to the users of those organizations who possess the certificate provided by CA. In case of the models in existence, it is mandatory for the user to reveal the credentials while attempting to access the VO resources which increases the probability the credentials being hacked. When the organizations encompassed in a VO are dynamic in nature and VO in itself is created in a dynamic manner, there arises a threat for the credentials of the user being exploited illegally. It is possible for an adversary user who has already hacked the user credential to easily misuse the resources of VO with the known credential that is revealed to the VOs. As a consequence, such systems are devoid of ensured security. Thus it is illustrated that the application of the existing models leaves the security of both the user and VO resources under constant threat.

In the proposed architecture we have attempted to access the resources with some keys and a token that are generated dynamically instead of signing in with its credentials for every access of the resources by the users. Since only request ids are passed to the VO there is no possibility of the user credentials being revealed to the VO. Also the model is designed in a way such that when a user tries to use the request id send by another user to access the resource, it is evaluated and rejected since the request id for each user is generated dynamically for every request. Thus the proposed architecture is simple and obviously more secure than previous approaches.

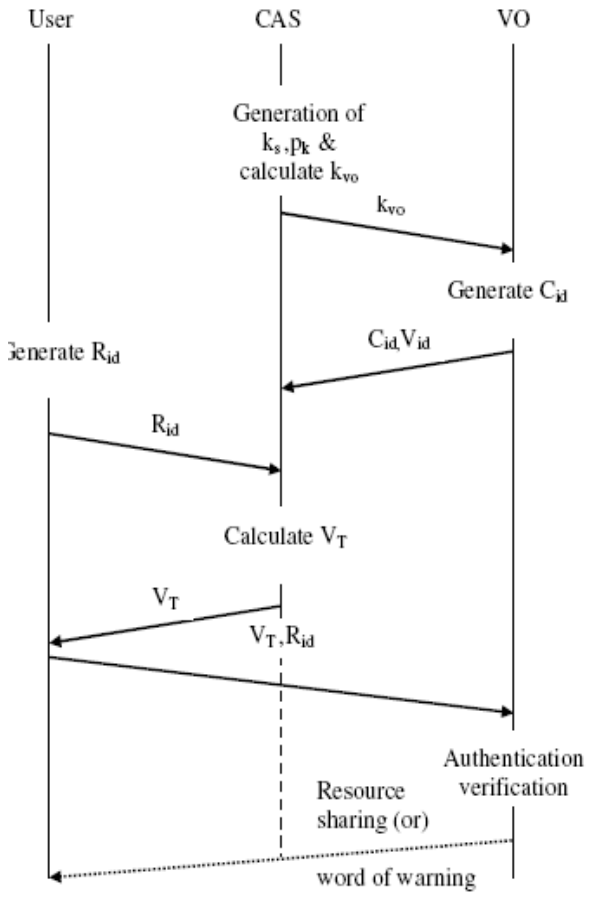

Fig. 1. Flow diagram of the proposed architecture

The concept of maintaining the user credentials as well as maintaining the organizational details in a role based grid computing VO organization is explained in this paper. The step by step flow of this process is as follows:

Step 1:

For every session or a particular period of time, CAS generates an arbitrary value p_k and its secret key k_s that is maintained by CAS to derive an intermediate key k_i which is given by

$$k_i = p_k^{k_s}$$

On basis of this, a VO key named as k_{vo} is calculated by multiplying k_s with the arbitrary value to the power of k_s - 1. This is represented by the equation

$$k_{vo} = p_k^{k_s - 1} * k_s \qquad \text{(I)}$$

CAS now sends k_{vo} to VO for further calculation that is to be performed later in this architecture.

Step 2:
VO stores k_s as the VO key through out the particular session. It then generates a confirmation id C_{id} in response to k_{vo} which will acknowledge the receipt of k_s. For every VO the C_{id} can be updated periodically. VO now sends its own VO id V_{id} and its corresponding C_{id} to CAS for validating the user, so that the CAS will hold V_{id} s for different VOs and its corresponding C_{id} s.

Step 3:
When a user from an organization in the VO needs to access the resources of the VO it sends a Request id R_{id} which is nothing but an arbitrary integer to CAS along with the certificate provided by CA.

Step 4:
After receiving the request from the user in terms of R_{id}, the CAS validates the identity certificate issued by the CA in order to determine if the user is valid one or not. Once the validation is done, CAS generates a validation token V_T for the respective user which also contains the role mentioned for the user by CAS. The value of V_T is provided by

$$V_T = k_s\left[1+\frac{C_{id}}{R_{id}}\left[1+\frac{k_i}{p_k}\right]-\frac{A_{id}}{R_{id}}\right] \qquad \text{(II)}$$

This I_T is multiplied with the k_s by the admin module of CAS and thus the final V_T is calculated. The value of V_T is provided by

$$V_T = k_s * I_T \qquad \text{(III)}$$

The admin sends this to the user directly since it does not want to reveal its private key to the proposed architecture module.

Step 5:
The user receives the V_T from the CAS and then sends it to the VO along with the R_{id} that was generated in step 4.

Step 6:
The Eventual Token validation process begins here. After the reception of the V_T and R_{id} from the user, the VO performs the Token validation as follows

$$\log[V_T]-\log\left[\frac{C_{id}}{R_{id}}k_{vo}\right]=0$$

When the above condition is satisfied, the VO will allow the user to access the resources. Otherwise a word of warning will be given to the adversary user.

4. Proof Of The Proposed Architecture

The entire flow of mechanism is validated by the following proof.

Let us assume that the aggregated id (A_{id}) is equal to the summation of Request id and Confirmation id.

$$(R_{id}+C_{id})=A_{id} \qquad (1)$$

Multiply with p_k on both sides of (1)

$$(R_{id}+C_{id})p_k = A_{id}\,p_k \qquad (2)$$

Add $C_{id}k_i$ on both sides of (2)

$$(R_{id}+C_{id})p_k + C_{id}k_i = A_{id}\,p_k + C_{id}k_i \qquad (3)$$

Substitute $k_i = p_k^{\,k_s}$ on both sides of (3)

$$(R_{id}+C_{id})p_k + C_{id}\,p_k^{\,k_s} = A_{id}\,p_k + C_{id}\,p_k^{\,k_s} \qquad (4)$$

Divide by p_k on both sides of (4)

$$R_{id}+C_{id}+\frac{C_{id}\,p_k^{\,k_s}}{p_k} = A_{id}+\frac{C_{id}\,p_k^{\,k_s}}{p_k} \qquad (5)$$

$$R_{id}+C_{id}\left[1+\frac{p_k^{\,k_s}}{p_k}\right] = A_{id}+\frac{C_{id}\,p_k^{\,k_s}}{p_k} \qquad (6)$$

Multiply with k_s on both sides of (6)

$$R_{id}k_s + C_{id}k_s\left[1+\frac{p_k^{\,k_s}}{p_k}\right] = k_sA_{id}+\frac{C_{id}\,p_k^{\,k_s}k_s}{p_k} \qquad (7)$$

$$k_s\left[R_{id}+C_{id}\left[1+\frac{p_k^{\,k_s}}{p_k}\right]\right]-k_sA_{id} = \frac{C_{id}\,p_k^{\,k_s}k_s}{p_k} \qquad (8)$$

Divide by R_{id} on both sides of (8)

$$k_s\left[\frac{R_{id}}{R_{id}}+\frac{C_{id}}{R_{id}}\left[1+\frac{p_k^{\,k_s}}{p_k}\right]-\frac{A_{id}}{R_{id}}\right] = \frac{C_{id}}{R_{id}}\frac{p_k^{\,k_s}k_s}{p_k} \qquad (9)$$

$$k_s\left[\frac{R_{id}}{R_{id}}+\frac{C_{id}}{R_{id}}\left[1+\frac{p_k^{\,k_s}}{p_k}\right]-\frac{A_{id}}{R_{id}}\right] = \frac{C_{id}}{R_{id}}\,p_k^{\,k_s-1}k_s \qquad (10)$$

Substitute $k_{vo} = p_k^{\,k_s-1}k_s$ in equation (10)

$$k_s\left[1+\frac{C_{id}}{R_{id}}\left[1+\frac{p_k^{\,k_s}}{p_k}\right]-\frac{A_{id}}{R_{id}}\right] = \frac{C_{id}}{R_{id}}k_{vo} \qquad (11)$$

Apply log on both sides of (11)

$$\log\left[k_s\left[1+\frac{C_{id}}{R_{id}}\left[1+\frac{p_k{}^{k_s}}{p_k}\right]-\frac{A_{id}}{R_{id}}\right]\right]-\log\left[\frac{C_{id}}{R_{id}}k_{vo}\right]=0 \quad (12)$$

When the aforesaid condition is satisfied we conclude that the details provided by the user are valid and access to the shared resources is provided to the user.

5. Experimental Results

The experimental results of the proposed approach are presented in this section. Our approach is programmed in JAVA V1.6. We have tested the proposed approach with a set of valid users and adversary users. Initially the user transmits a request id to CAS and CAS verifies this key with the confirmation key obtained from VO. CAS then provides Validation token which is used to validate the user. The inputs from two different user classifications are evaluated and the respective results at every stage are represented in the tabular column provided below.

The tabular column given below shows the validation of the users by verifying the validation code and thus it identifies whether the user is valid or not. Some examples for valid users are given below in this table.

Sl. No.	Arbitrary Value p_k	Secret Key k_s	V O Key k_{vo}	Intermediate Key k_s	Request Id R_{id}	Confirmation Id C_{id}	Aggregated Id A_{id}	$\log V_T$	$\log\left[\frac{C_{id}}{R_{id}}k_{vo}\right]$	Validation: Is User Valid
1	2.0	9.0	512.0	2304.0	84.0	34.0	118.0	6.837946	6.837946	Valid User
2	1.0	145.0	1.0	145.0	11.0	13.0	24.0	5.143788	5.143788	Valid User
3	7.0	149.0	8.310132762606204E125	1.768871116611892E127	53.0	14.0	67.0	291.66742	291.66742	Valid User
4	1.0	237.0	1.0	237.0	70.0	87.0	157.0	5.685473	5.685473	Valid User
5	6.0	66.0		2.5082753518537416E52	74.0	25.0	99.0	119.56883	119.56883	Valid User

A user is identified to be invalid when the validation fails. The table provided below clearly shows that the validation code has not been balanced and thus the user is evaluated as an invalid user. Two examples for invalid users are given in this table.

Sl. No	Arbitrary Value p_k	Secret Key k_s	V O Key k_{vo}	Intermediate Key k_i	Request Id R_{id}	Confirmation Id C_{id}	Validation Id V_{id}	$\log V_T$	$\log\left[\frac{C_{id}}{R_{id}}k_{vo}\right]$	Validation: Is User Valid
1	7.0	35.0	3.788186922656648E29	1.8940934613283236E30	66.0	87.0	153.0	69.85147	69.992546	Invalid User
2.	1.0	151.0	1.0	151.0	12.0	40.0	52.0	3.8233573	6.2212524	Invalid User

6. Conclusion

Role based access control in grid computing for VOs is beneficial since the organizations possess a dynamic computing process. In this paper we have presented a novel security framework, for role based access controlled organizations, that makes a secure environment for this computing process. As the user, the CAS and the VO hold all the evaluation mechanisms within themselves and only very few values are exchanged between them, the possibilities for hacking any of these values are very tedious. Dynamic generation of all the evaluation mechanisms adds additional advantage. Moreover it is not necessary for the user to provide any of its credentials to VOs which are involved in the access control and this makes our framework strong. Thus credential details about the user are maintained in a secured manner. This framework had been implemented in JAVA V1.6 and we have visualized its performance by employing some valid users and adversary users. Hence this well-built frame work offers a secure atmosphere for role based control in grid computing without taking any of the credential of the user into consideration thereby avoiding the menacing effects caused by the accessing of resources by adversary users besides maintaining the user's credentials confidentially.

References

[1] I. Foster, C. Kesselman, "The Grid: Blueprint for a new Computing Infrastructure", Morgan Kaufmann Publishers, ISBN 1-55860-475-8, 1999.

[2] I. Foster, C. Kesselman, and S. Tuecke. "The Anatomy of the Grid: Enabling Scalable Virtual Organizations". International Journal of High Performance Supercomputing Applications, vol: 15, no: 3, 2001.

[3] Miguel L. Bote-Lorenzo, Yannis A. Dimitriadis, Eduardo Gmez-Snchez, "Grid Characteristics and Uses: A Grid Definition", In European across Grids Conference, Santiago de Compostela, Spain, pp: 291-298, 2003.

[4] Marty Humphrey, Mary R Thomson and Keith R Jackson, "Security for Grids", Proceedings of the IEEE, Vol 93, No 3, pp: 644-652, 2005.

[5] Von Welch, Frank Siebenlist, Ian T. Foster, John Bresnahan, Karl Cza-jkowski, Jarek Gawor, Carl Kesselman, Sam Meder, Laura Pearlman, Steven Tuecke, "Security for Grid Services", In 12th International Symposium on High-Performance Distributed Computing, Seattle, WA, USA, pp: 359-368, 22-24, June 2003.

[6] Jiri Denemark, Michał Jankowski, Ludek Matyska, Norbert Meyer, Miroslav Ruda, Pawel Wolniewicz, "User Management for Virtual Organizations", CoreGRID Integration Workshop Proceedings, Pisa 2005.

[7] Glen Wasson and Marty Humphrey, "Policy and Enforcement in Virtual Organizations", In 4th International Workshop on Grid Computing (Grid2003) Phoenix, AZ., pp: 125-133, 2003.

[8] Cui-xiao Zhang, Ying-xin Hu, Guo-bing Zhang, "Task-Role Based Dual System Access Control Model", IJCSNS International Journal of Computer Science and Network Security, Vol.6, No.7, pp: 211-215, July 2006.

[9] Sandhu, R., Ferraiolo, D., and Kuhn, R. "The NIST model for role-based access control: Towards a unified standard". In Proceedings of 5th ACM Workshop on Role- Based Access Control, pp: 47-63, 2000.

[10] Ganesh Godavari and Edward Chow, "Secure Information Sharing Using Attribute Certificates and Role Based Access Control", In Proceedings of Security and Management'2005, pp: 269-276, 2005.

[11] Uday O. Ali Pabrai, "HIPAA Security and Role Based Access Control (RBAC),"HIPAA Academy, September 5, 2003.

[12] R.S. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. "Role-based access control models", IEEE Computer, Vol: 29, No: 2, pp: 38–47, 1996.

[13] Benjamin Aziz, Simon N. Foley, John Herbert, Garret Swart, "Reconfiguring Role Based Access Control Policies Using Risk Semantics", Journal of High Speed Networks, Vol: 15, No: 3, pp: 261-273, 2006.

[14] Globus Project. Globus Project Web Site, 2009. http://www.globus.org/.

[15] Anil L. Pereira, Vineela Muppavarapu, and Soon M. Chung, "Role-Based Access Control for Grid Database Services Using the Community Authorization Service", IEEE Transactions On Dependable And Secure Computing, Vol. 3, No: 2, 2006.

[16] J.B.D. Joshi, R. Bhatti, E. Bertino, and A. Ghafoor, "Access-Control Language for Multi-domain Environments," IEEE Internet Computing, Vol. 8, No. 6, pp. 40-50, Nov.-Dec. 2004.

[17] T. Mayfield, J.E. Roskos, S.R. Welke, and J.M. Boone, "Integrity in Automated Information Systems," technical report, Nat'l Computer Security Center, pp: 79-91, 1991

[18] Jongil Jeong, Weehyuk Yu, Dongkyoo Shin, Dongil Shin, Kiyoung Moon, and Jaeseung Lee, "Integration of Single Sign-On and Role-Based Access Control Profiles for Grid Computing", 8th Asia-Pacific Web Conference on Frontiers of WWW Research and Development , Vol: 3841, pp. 880-885, January 16-18, 2006.

[19] Lorenzo Cirio, Isabel F. Cruz, and Roberto Tamassia, "A Role and Attribute Based Access Control System Using Semantic Web Technologies", In Int. IFIP Workshop on Semantic Web and Web Semantics, volume 4806 of Lecture Notes in Computer Science, pp: 1256-1266. Springer, 2007.

[20] Chen Ying, Yang Shoubao, Guo Leitao, Liu Pengzhan, Shen Kai, "Design and Implementation of Dynamic-Role Based Access Control Framework in Grid Environment", International Conference on Information Technology: Coding and Computing, Vol: 2, pp: 758- 759, 2005.

[21] Nathan Dimmock, Andras Belokosztolszki, David Eyers, Jean Bacon, Ken Moody, "Using Trust and Risk in Role-Based Access Control Policies", Proceedings of the ninth ACM symposium on Access control models and technologies, pp: 156 - 162, 2004.

[22] G Geethakumari, Atul Negi, V.N Sastry, "A Cross - Domain Role Mapping and Authorization Framework for RBAC in Grid Systems", IJCSA, Vol: 6, No: 1, 2009.

M.Nithya obtained B.E. Degree in 2003 from Periyar University with First Class and her M.E. Degree in 2005 from Anna University with First Class. She had presented many papers on Securities in Grid computing also she had attended many seminars and conferences on Grid Computing. With good knowledge on Area of Grid Computing she is now undergoing her full time research in "GRID SECURITY" under Anna University, Chennai for her PhD research. Some of her presentation titles are "Use of Grid Computing", "GRID Security using Web Services", etc.

R.S.D.Wahida Banu obtained B.E. degree in 1981 and her M.E. degree in Jan '85 from GCT, Coimbatore, and Madras University. She got the Ph.D. degree from Anna University, Chennai. First lady to acquire Ph.D. in Chennai zone and second qualified Ph.D. supervisor in the area of Computer Science and Engineering related areas. As expertise is less it continues in the Directorate of Technical Education, Tamil Nadu. She has more than 25 years of Teaching Experience. She is the member of ISOC, IAENG, VDAT and life member of ISTE, IE, CSI and SSI. She has published more than 100 National and International Journals. She has produced 5 PhD Scholars. She is currently working as Professor and Head of Electronics and Communication Engineering, Government College of Engineering, Salem. Her area of interest includes Artificial Intelligence, Network Security and Grid Computing.