International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 1, January 2014)

An Empirical Study of Security Issues in Grid Middleware

Rakesh Bhatnagar[1], Dr. Jayesh Patel[2]

[1]*Asst. Prof. & Coordinator (MCA Department), S. K. Patel Institute of Management & Computer Studies, Gandhinagar-382023, India*

[2]*Associate Prof., Acharya Motibhai Patel Institute of Computer Studies, Ganpat University, Kherva-382711, India*

***Abstract*-- With the passing of each day, Grid computing and Cloud computing are becoming an important integral part of various applications. More and more software professionals and corporate are focusing on these high performance computing mechanisms. These mechanisms are tremendously evolving as users are also continuously striving for smarter applications and solutions. And as these mechanisms are evolving, their development environment, architecture, middleware and security aspects are also evolving.**

Security is now became an area of concern for all business applications. The information security issues are there all over the field of Grid computing. One of the keys to survive new and emerging threats is by providing policy based access and context sensitive access to the users. Businesses should ensure that they only give the right access to the right people and they should get the information on who is on their network and what they are doing.

This paper discusses and reviews various properties, characteristics, and challenges of the security middleware approaches available. The review reveals that many options of security in middleware are available but there are still many issues and challenges yet to be solved. The information provided in this paper can be used by those organizations which needs proper security mechanism in distributed communication so that a strong security model can be devised further. This paper also puts a challenge to us to devise proper security policies for such communication protocols.

***Keywords*-- Authentication, Credential Management, Single sign-on, X.509 certificates**

I. Introduction

A Grid environment is a distributed environment having lot of diverse components used for various applications. These diversified components contain all the software, hardware, services and resources needed to execute the applications.

There are three core areas of a Grid environment – Architecture, Middleware and Services. In this paper, we have focused on the Middleware. Middleware exists on top of the operating system and helps in assisting & connecting software applications and components.

There are several middleware that are used for different types of application. Each middleware has different communication policies & rules as well as different modes of operation. Based on this middleware can be classified into procedure oriented, object oriented, message oriented, component based or reflective [1].

The functionalities of the middleware can be again classified into following categories – application specific, management specific and information exchange specific.

The application-specific middleware provides services for various types of applications such as distributed transaction processing, distributed-database services, specialized services for mobile computing and multimedia etc. The management middleware is used to manage security, communicate with servers, handle failures, and monitor performance. The information-exchange middleware is used for information management and communication of information to different places.

The problem with the GRID environment is its diversity of the distributed components and their large number of users. These can make them vulnerable to faults, failure and excessive loads. Thus, security becomes a very important issue because most transactions and operations involve online data transmission. It is very important to protect the applications and the data involved from malicious, unauthorized and unintentional attacks. To solve this issue and provide security, there should be well defined access policies, cryptography (i.e. encryption and decryption) mechanisms and authentication models.

Middleware is used to provide integration and interoperability among the distributed components in the grid environment. So, it can help in providing the security needed for the grid environment. Security components lie in the layers below the client application in the architecture to maintain strong abstraction, portability, efficiency and automation. These security components include access control and audit policies that help in achieving interoperability.

In this paper, different security middleware and their approaches are discussed and summarized which help to provide different security measures for applications.

II. Characteristics Of Grid Computing Environment

We take a hypothesis of a multi-institutional academic collaboration. Suppose a faculty, who is a member of an academic organization or university, receives a request from a colleague regarding updated syllabus copy and new list of practical programs of the subject that he is going to teach.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 1, January 2014)

Faculty use grid computing to furnish the request. For this, he needs to start a program that will send code to the remote location where the data i.e. list of practical programs, is stored say place A. Next for analytical reasons, the program runs a simulation so that results can be compared. For running the simulation, a resource broker service is contacted that is maintained by the organization or university at place B. This service is used to locate idle resources in the organization that can be used for running the simulation. The resource broker now starts computation on computers at two places say C and D. These computers then access some parameter values that are stored on a file system at place say E. This communication is depicted in Figure 1.

This example highlights many distinctive characteristics of the grid computing environment:

- Dynamic and Large set of Users – members of various colleges of the organization can take part in the communication.
- Dynamic and Large set of Resources - the quantity of resources and location can change rapidly because individual institutions and users can decide whether and when to use and allow the resources.
- Dynamic and Large set of Processes – dynamic processes will be running with various resources and at various sites.
- Communication Mechanism – Different communication mechanism like unicasting, multicasting or broadcasting can be used by processes.
- Authentication & Authorization Policies - proper authentication and authorization mechanisms and policies are required such as SSL, SSH, X.509 certificates, cryptography methods etc.
- Credential Management – Each user should be assigned a unique credential (UserID & password). Access through guest account should be properly managed.
- Diversified resources & users – Users and Resources involved in the communication may be present at different colleges or institutions.

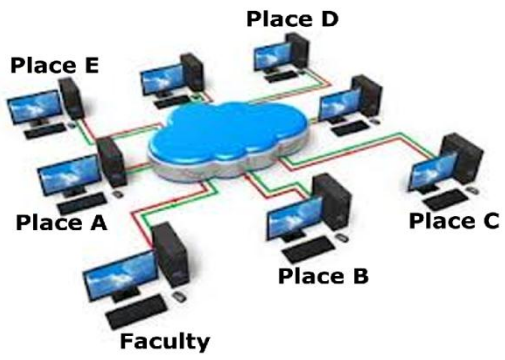

Figure 1: Communication in Academic Organization / University

III. Grid Middleware Approaches

We have surveyed different grid middleware and their security solutions. The objective was to find out what is presently available and what is needed to improve the available security solution. We also focused on how efficient is the middleware in terms of its security solution.

1. UNICORE [2]

Security is maintained in the UNICORE system by single sign-on. For this it uses X.509 version 3 certificates.

Prior to use the system, the user must configure their client to use their digital certificate, imported into the Java file called as "keystore". The user must also import the CA - certificate authority certificates for the resources that they want to access and use. When the client is started, the user has to give their password to unlock the "keystore" file. After this, when the client submits a job to the system, it uses the certificate to digitally sign the AJO (Abstract Job Object), that are the collection of java classes, before it is transmitted to the NJS (Network Job Supervisors) that is the server. After this the signature is verified using a copy of the user's certificate maintained at the server, If verified, the identity of the user is established.

The X.509 certificate [2] works as follows – Asymmetric key algorithms of cryptography are used in which the encryption and decryption keys are different. Thus it uses public key encryption mechanism. This set up is also called PKI (Public Key Infrastructure).

Two things can be ensured in UNICORE with X.509 certificates [2]:

- Each client or server can prove its identity. It is done by presenting its certificate containing the public key and providing evidence of the private key.
- Encrypted messages can be read by private key only. This way an encrypted communication channel between different users on the Grid is established. The protocol uses the Secure Sockets Layer (SSL) mechanism.

2. GLOBUS [3]

Globus also uses X.509 certificates for providing security. In Globus, a user has to convert his digital certificate to an enhanced email composition. Then he applies the correct file permissions to it and keeps it into a specified folder in the home directory. Before using any command, users have to create a proxy version of their certificate which allows grid middleware tools to perform actions on their behalf. Then after this, the client tools invoke a mutual authentication protocol to establish the user's identity via the proxy certificate.

3. Sun Grid Engine [4]

In Sun Grid Engine, secret key encryption is used for the messages. The public key algorithm is used to exchange the private key. The user has to present certificate to prove identity, and in response he receives a certificate from the system. This ensures that the communication is correct and valid. This establishes a session for communication. After this, the communication continues in encrypted form only. Being valid only for a limited and certain period, the session ends accordingly. To continue the communication, session has to be reestablished.

4. Alchemi [5][6]

Alchemi middleware system uses a role-based authentication security model, a thread model to execute and submit jobs and a cross platform web service model for interoperability. Jobs are represented as XML files. Base64 encoding decoding is used. But it does not support single sign-on and delegation mechanism. With the use of X.509 proxy certificates, these mechanisms are implemented on the Alchemi framework. In order to perform this implementation, a trusted communication is set up between two hosts. Then the client host will be authenticated by the server host. After authentication, Client gets access of various applications provided by the framework with full delegation rights.

5. HTCondor [4][7]

In HTCondor (High throughput computing Condor) which was formerly known as Condor, the security is applied through authentication mechanism by using authentication methods like GSI authentication, Windows authentication, Kerberos authentication, file system authentication etc. Users use public key certificates in GSI authentication as a result problem of identity spoofing persists. Checkpoint servers are also used but they can be replaced maliciously resulting in poor retrieval of data. In Condor system, the problem of DoS persists that is "Denial of Service" if proper access permissions are not set. Even the configuration files and log files can be changed. The user gets full access of the job once he submits it. This means, he even can change the owner of the job.

6. Entropia [4][8]

A range of encryption and sandboxing technologies are used in the Entropia security service. Through sandboxing, a single machine virtualization is provided. This helps in enforcement of security policies and management of identity on a single application and the resources. Security is provided by providing cryptography on the applications that communicate and allocating access rights of file system, the interface etc to the jobs. But this increases the cost and reduces the high performance capability of the system.

7. Xgrid [4][9]

Single sign-on is permitted in Xgrid. This is done through Kerberos. For establishing a connection, a password is required to be sent by the client to the server for executing a job. In response, the server again uses a password for authentication. The encryption used is MD5. For different users, different permissions are set. No certificate authentication or any other mechanism is used. Problem of identity spoofing persists in Xgrid.

8. iDataGuard [10]

It is an interoperable security grid middleware. Users are allowed to outsource their file system to heterogeneous data storage providers such as Rapidshare and Amazon. It uses cryptographic techniques to maintain data integrity and confidentiality. Normally data at client side iss stored in plaintext and the cryptography is applied on the storage provider. Due to this, outside and inside attacks can be possible. iDataguard solves this problem by applying cryptography. It also provides index based search of keyword. But certificate authentication is not supported.

9. Middleware for Mobile Computing

A survey of middleware paradigms for mobile computing [12] presented several research projects that have been started to address the dynamic and security aspects of mobile distributed systems. It is clear from the paper that traditional middleware approaches based on technologies like CORBA [12] and Java RMI [12] which support object oriented middleware are successful in providing heterogeneity and interoperability but they are not able to provide the appropriate and required support for modern and advanced mobile computing applications.

10. Selma [15]

One of the mobile agent based middleware is Selma. This middleware provides neighborhood discovery and wireless communication for distributed application in mobile multiple ad-hoc networks. Traditional technologies like CORBA and Java RMI are not suitable for middleware supporting mobile ad-hoc networks. In mobile network based middleware there will be multiple nodes distributed across the network. All of these nodes can be accessed through the above technologies. The paper is concluded by arguing that a complete middleware solution still do not exists that provides and fulfills all the required functionalities for a middleware concerning security aspects and resource management. No certification is used for the authentication.

11. XtremWeb [16]

It is a open source middleware for peer to peer distributed grid applications and desktop grids. In this, any participant can be a user. The access rights provided are UNIX based and sandboxing technology is also used.

It uses public key algorithm for authentication. Problem in this is the grid security policy is breached as the pilot job owner i.e. the one who started the job can be different from the final job owner.

IV. FINDINGS

From the survey of all the above middleware systems, it is clear that many security parameters are addressed by them and many are still pending. We also came to know that the security attacks and security breaches can be of many types. They are given below -

- Eavesdropping: In this, the information that is transferred can be tapped without affecting system resources. This can increase the network traffic.
- Replay attacks: In this, the information that is transferred is captured and reproduced.
- Access control attacks: In this, a person gets unauthorized access to resources as the access rights are not defined precisely.
- Masquerading: In this, during a communication one suspicious grid entity pretends as another valid grid entity and it results in the disclosure of confidential information and unauthorized transactions.
- Brute force attacks: In this, a suspicious person i.e. the attacker tries to decipher the encrypted data, until a legible plaintext is obtained.

Issues of Security are summarized as follows -

- Naming of rights by servers
- No idea of number of servers needed to complete the task
- Improper Authentication management
- Improper authorization and access control policies
- Data sharing and Integrity violation
- Open communication protocols
- Weak trust relationship policies
- Inability of working with less resources
- Single sign on mechanism
- Proper credential management
- Interoperability with local security solutions
- Inter-domain Exportability
- Capability for multiple implementations

Consequences of these issues are -

- Complete functionalities are not provided
- Desired results are not satisfactory
- Redundancy in functionalities of applications may occur which would waste time and effort.
- Scalability of applications – when applications increase in size and functionality, it will be difficult to send them to smaller devices.
- Interoperability becomes complex and difficult to achieve.

The list is endless.

V. CONCLUSION

We studied various security mechanisms of different middleware and found that even though, most projects succeeded in implementing their proposed security middleware, it is not easy to provide a fully-secured middleware that can combine all security components in a one-stop solution and can handle all the security attacks. There is no single security middleware that could claim full protection against the possible security attacks and the risks. One main reason for this is that application requirements vary drastically depending on the domain and operating environment.

Creating security policies by taking proper care of all the issues listed above will result in achieving better security.

VI. FUTURE PROSPECTS

Considering the information available through this paper, a framework can be developed that can provide application specific, management specific and information specific security in the middleware systems.

REFERENCES

[1] Middleware, retrieved Jan 10, 2014, from: *www.cl.cam.ac.uk/teaching/1011/CDSysII/12-middleware.pdf*

[2] Unicore Client user manual retrieved Jan 10, 2014, from: *www.unicore.eu/documentation/manuals/unicore6/RichClient-6.3.1.pdf*

[3] NV Kanaskar, U Topaloglu, C Bayrak, Globus security model for grid environment, ACM SIGSOFT Software Engineering Notes, 2005 - dl.acm.org, Volume 30 Issue 6, November 2005 Pages 1 - 9

[4] Syed Ishtiaque Ahmed, Mustafizur Rahman and Mukaddim Pathan, Policy Requirements to Ensure Security in Enterprise Grid Security Systems,2nd International Conference on Computer Science and its Application, 2009

[5] Manisha Bhardwaj, Sarbjeet Singh & Makhan Singh, Implementation of Single Sign-on and Delegation mechanism in Alchemi.Net based Grid Computing Framework, International Journal of Information Technology and Knowledge Management January-June 2011, Volume 4, No. 1, pp. 289-292

[6] Akshay Luther, Rajkumar Buyya, Rajiv Ranjan, and Srikumar Venugopal, Alchemi: A .NET-based Enterprise Grid Computing System, Internet Computing, 2005 - cloudbus.cis.unimelb.edu.au

[7] J Frey, T Tannenbaum, M Livny, I Foster and S Tuecke, Condor-G: A computation management agent for multi-institutional grids, Cluster Computing, 2002 – Springer

[8] A Chien, B Calder, S Elbert, K Bhatia, Entropia: architecture and performance of an enterprise desktop grid system, Journal of Parallel and Distributed Computing, Volume 63, Issue 5, May 2003, Pages 597–610, Elsevier

[9] Marcos Dias de Assuncao, Krishna Nadiminti, Srikumar Venugopal, Tianchi Ma, Rajkumar Buyya, An Integration of Global and Enterprise Grid Computing: Gridbus Broker and Xgrid Perspective, Grid and Cooperative Computing - GCC 2005, Springer, Lecture Notes in Computer Science Volume 3795, 2005, pp 406-417

[10] R.C. Jammalamadaka, R. Gamboni, S. Mehrotra, K. Seamons, N. Venkatasubramanian, iDataGuard: Middleware providing a secure network driven interface to untrusted internet data storage, in Proc. International Conference on Extending Database Technology: Advances in DB Technology, Nantes, France, in: ACM International Conference Proceedings Series, vol. 261, 2008.

[11] N. Zhang, J. Chin, A. Rector, C. Goble, Y. Li, Towards an authentication middleware to support ubiquitous Web access, in Proc. International Computer Software and Applications Conference, COMPSAC, vol. 2, September 2004, pp. 36_38.

[12] A. Gaddah, T. Kunz, A survey of middleware paradigms for mobile computing, Technical Report SCE-03-16, Carleton Univ. Systems and Computing Engineering, 2003.

[13] F. Almenarez, A. Marin, A.D. Diaz, A. Cortes, C. Campo, C. Garcia-Rubio, A trust-based middleware for providing security to ad-hoc peer-to-peer applications, in Proc. 6th IEEE International Conference on Pervasive Computing and Communications, PerCom 2008, pp. 531_536.

[14] S.S. Yau, X. Zhang, A middleware service for secure group communication in mobile ad hoc networks, in Proc. International Computer Software and Applications Conference, COMPSAC, 2003, pp. 10_15.

[15] Daniel Gorgen, Hannes Frey, Johannes K. Lehnert & Peter Sturm, SELMA: A middleware platform for self-organizing distributed applications in mobile multihop ad-hoc networks, Simulation MultiConference WMC, 2004

[16] Gabriel Caillat, Oleg Lodygensky, Etienne Urbah, Gilles Fedak and Haiwu He, Towards a Security Model to Bridge Internet Desktop Grids and Service Grids, Euro-Par 2008 Workshops - Parallel Processing, Lecture Notes in Computer Science Volume 5415, 2009, pp 247-259, Publisher Springer

[17] S.I. Ahamed, M. Haque, K.M. Asif, S-MARKS: A middleware secure by design for the pervasive computing environment, in Proc. of 4th International Conference on Information Technology: New Generations, ITNG, 2006, IEEE CS Press, 2007, pp. 303_308.

[18] J. Hightower and G. Borriello, Location systems for ubiquitous computing, IEEE Computer, vol. 34, pp. 57_66, August 2001.

[19] M. Mauve, J. Widemer, and H. Hartenstein, A survey on position-based routing in mobile ad-hoc networks, IEEE Network Magazine, vol. 15, no. 6, pp