An approach for security in grid computing on example.com web server

[1]Dinesh Yadav, [2]Deepak Bhatnagar

Abstract---Grid computing has arisen as an evolution of distributed systems mainly focused on the sharing of and remote access to resources in a uniform, transparent, secure, efficient and reliable manner. It is possible to join Grid methodology and web technology in order to create one of the most promising technologies and developments to appear in recent years, in that they enrich one another and provide new solutions that solve many of the limitations and problems found in different technologies.

In this paper we focus on Security in grid computing. In the manner of security we apply authentication on grid server that hold authentication information about each client's login. So on the basis of login, User Enabled Collaboration and IP authentication, we find out valid or invalids user.

Index Terms— Grid computing, protocol, TCP/IP layer, packet tracer simulator.

1 INTRODUCTION

Large-scale distributed computing environments, or "computational grids" as they are sometimes termed [4], couple computers, storage systems, and other devices to enable advanced applications such as distributed supercomputing, teleimmersion, computer-enhanced instruments, and distributed data mining [2]. Grid applications are distinguished from traditional client-server applications by their simultaneous use of large numbers of resources, dynamic resource requirements, use of resources from multiple administrative domains, complex communication structures, and stringent performance requirements, among others. While scalability, performance and heterogeneity are desirable goals for any distributed system, the characteristics of computational grids lead to security problems that are not addressed by existing security technologies for distributed systems. For example, parallel computations that acquire multiple computational resources introduce the need to establish security relationships not simply between a client and a server, but among potentially hundreds of processes that collectively span many administrative domains. Furthermore, the dynamic nature of the grid can make it impossible to establish trust relationships between sites prior to application execution. Finally, the inter domain security solutions used for grids must be able to interoperate with, rather than replace, the diverse intra-domain access control technologies inevitably encountered in individual domains.

2 GRID SECURITY PROBLEM

We introduce the grid security problem with an example illustrated in Figure 1. This example, although somewhat contrived, captures important elements of real applications.

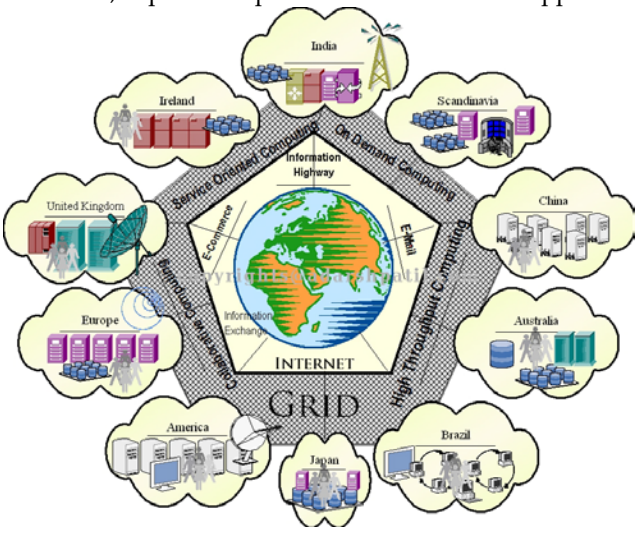

Figure 2 Architecture of the Grid Security

We imagine a scientist, a member of a multi-institutional scientific collaboration, who receives e-mail from a colleague regarding a new data set. He starts an analysis program,

- [1]*Dinesh Yadav is currently pursuing masters degree program in computer science and engineering from college of science and engineering, Jhansi, in Uttar Pradesh Technical University, India*
  E-mail: dineshyadav14@gmail.com
- [2]*Deepak Bhatnagar is currently head of department of computer science and engineering in college of science and engineering ,Jhansi ,India,*
  E-mail:bhatnagar.deepak@rediffmail.com

This dispatches code to the remote location where the data is stored (site C). Once started, the analysis program determines that it needs to run a simulation in order to compare the experimental results with predictions. Hence, it contacts a resource broker service maintained by the collaboration (at site D), in order to locate idle resources that can be used for the simulation. The resource broker in turn initiates computation on computers at two sites (E and G). These computers access parameter values stored on a file system at yet another site (F) and also communicate among themselves (perhaps using specialized protocols, such as multicast) and with the broker, the original site, and the user.

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

1. The user population is large and dynamic. Participants in such virtual organizations as this scientific collaboration will include members of many institutions and will change frequently.
2. The resource pool is large and dynamic. Because individual institutions and users decide whether and when to contribute resources, the quantity and location of available resources can change rapidly.
3. A computation (or processes created by a computation) may acquire, start processes on, and release resources dynamically during its execution. Even in our simple example, the computation acquired (and later released) resources at five sites. In other words, throughout its lifetime, a computation is composed of a dynamic group of processes running on different resources and sites.
4. The processes constituting a computation may communicate by using a variety of mechanisms, including unicast and multicast. While these processes form a single, fully connected logical entity, low-level communication connections (e.g., TCP/IP sockets) may be created and destroyed dynamically during program execution.
5. Resources may require different authentication and authorization mechanisms and policies, which we will have limited ability to change. In Figure 1, we indicate this situation by showing the local access control policies that apply at the different sites. These include Kerberos, plaintext passwords, Secure Socket Library (SSL), and secure shell.
6. An individual user will be associated with different local name spaces, credentials, or accounts, at different sites, for the purposes of accounting and access control. At some sites, a user may have a regular account (\ap," \physicist," etc.). At others, the user may use a dynamically assigned guest account or simply an account created for the collaboration.
7. Resources and users may be located in different countries.

3 GRID SECURITY POLICY

Before delving into the specifics of security architecture, it is important to identify the security objectives, the participating entities, and the underlying assumptions. In short, we must define a security policy, a set rules that define the security subjects (e.g., users), security objects (e.g., resources) and relationships among them. While many different security policies are possible, we present a specific policy that addresses the issues introduced in the preceding section while reflecting the needs and expectations of applications, users, and resource owners. To our knowledge, the following discussion represents the first such grid security policy that has been defined to this level of detail.

In the following discussion, we use the following terminology from the security literature:

1. A subject is a participant in a security operation. In grid systems, a subject is generally a user, a process operating on behalf of a user, a resource (such as a computer or a file), or a process acting on behalf of a resource.
2. A credential is a piece of information that is used to prove the identity of a subject. Passwords and certificates are examples of credentials.
3. Authentication is the process by which a subject proves its identity to a requestor, typically through the use of a credential. Authentication in which both parties (i.e., the requestor and the requested) authenticate themselves to one another simultaneously is referred to as mutual authentication.
4. An object is a resource that is being protected by the security policy.
5. Authorization is the process by which we determine whether a subject is allowed to access or use an object.
6. A trust domain is a logical, administrative structure within which a single, consistent local security policy holds. Put another way, a trust domain is a collection of both subjects and objects governed by single administration and a single security policy.

4 PROPOSED METHODOLOGY

An Ethernet based LAN with six Pentium Machines is used. All of these machines are used as Client with Windows XP. All machines connected with different-2 servers by routers. Below figure showing the grid based network diagram.

Configuration of The Model :-

PC Clients- 1
PDA - 3
Tablets - 3
Router - 3
DNS Server- 2
Grid Server- 2
Root Server- 1
Authentication Server- 1

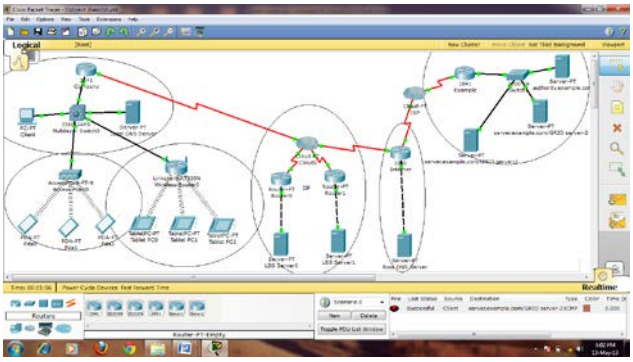

Fig.4.1

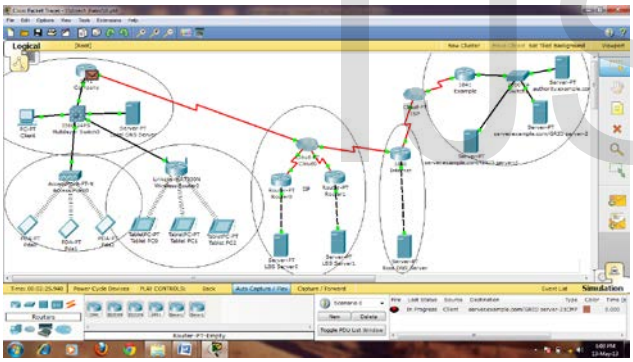

Fig 4.2

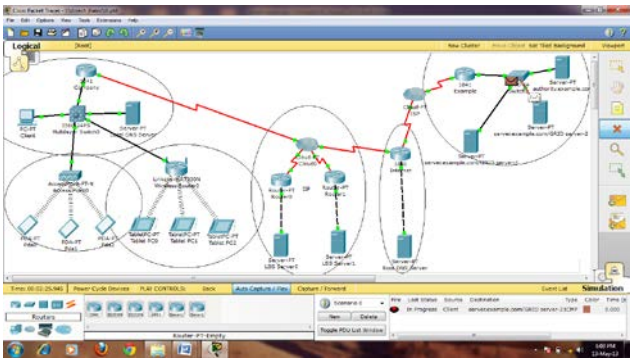

Fig. 4.3

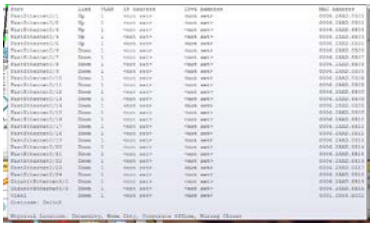

Fig 4.4

Fig. 4.5

5 CONCLUSION

Our research work has consistently pointed security, this potential problem for grid computing. The following different solutions implemented on our model provide security solutions for different areas – Firewalls, authentication and User Enabled Collaboration Mechanism using Security, and Authentication. The combinations of all solutions are much better than previous solution for security in grid computing.
Within the implementation, the use of packet tracer network simulator provides for portability. Group communication is one major requirement not addressed. The security design presented addresses a number of scalability issues. The sharing of credentials by processes created by a single resource allocation request means that the establishment of process credentials will not, we expect, be a bottleneck. The fact that all resource allocation requests must pass via the user proxy is a potential bottleneck; this must be evaluated in realistic applications and, if required, addressed in future work. One major scalability issue that is not addressed is the number of users and resources. However, we believe the current approach can deal with this.

6 FUTURE WORK

The future research work in this area is endless. Still there are several points which need to be answer by researchers. Some of the points on which research work could be carried out are listed below:

1. In this work, Grid Web server is optimized by using the LA taken from various Grid Web servers. In future, hardware of various Grid Web servers will be taken into account for optimization of Grid Web servers.

2. In future, bandwidth optimization will be taken into account so that, the process of Grid Web server optimization will be reach to its epic point.
3. The concept of fault tolerance will be more advance in future in order to optimize the performance of Grid Web servers.
4. Appropriate solution for other services such as File Transfer Protocol (FTP) or proprietary protocols can be focused.

REFERENCE

[1] W. Kanoun, N. Cuppens-Boulahia, F. Cuppens, S. Dubus,and A. Martin, "Success likelihood of ongoing attacks for intrusion detection and response systems," in Proceedings
of the 2009 International Conference on Computational Science and Engineering - Volume 03. Washington, DC, USA: IEEE Computer Society, 2009, pp. 83–91.
[Online]. http://portal.acm.org/citation.cfm?id=1632709.1633494

[2] Y. Xiang and W. Zhou, "Protect grids from ddos attacks," in GCC, 2004, pp. 309–316.

[3] "Scalable simulation framework (ssf): A public-domain standard for discrete-event simulation of large, complex systems in java and c++," 2010.
[Online]. http://www.ssfnet.org/homePage.html.

[4] V. Welch, J. Gawor, C. Kesselman, S. Meder, and L. Pearlman, "Security for grid services," in Twelfth International Symposium on High Performance Distributed Computing (HPDC-12). IEEE Press, 2003, pp. 48–57.

[5] I. Foster, C. Kesselman, J. M.Nick, and S. Tuecke, "The physiology of the grid: An open grid services architecture for distributed systems integration," 2002.
[Online].
http://www.globus.org/alliance/publications/papers/ogsa.pdf

[6] "The open grid services architecture, version 1.5," Available from: http://www.ogf.org/documents/GFD.80.pdf, 2002-2006.

[7] R. Ford, M. Bush, and A. Bulatov, "Predation and the cost of replication: New approaches to malware prevention?" Computers & Security, vol. 25, no. 4, pp. 257–264, 2006.