International Journal of Information Technology and Knowledge Management
January-June 2011, Volume 4, No. 1, pp. 289-292

IMPLEMENTATION OF SINGLE SIGN-ON AND DELEGATION MECHANISMS IN ALCHEMI.NET BASED GRID COMPUTING FRAMEWORK

Manisha Bhardwaj[1], Sarbjeet Singh[2] & Makhan Singh[2]

Grid computing is a new paradigm for accessing resources distributed across organizational boundaries using computer networks. It allows for new ways of accessing & using remote resources. This new paradigm demands for several new concepts concerning security, especially authentication and authorization. The most commonly used security framework for grid system is Grid Security Infrastructure. It makes use of proxy certificates & private keys as credentials for allowing single sign on and delegation of rights. Single Sign-On (SSO) are the desired feature of computational grids. Although there are several ways to implement single sign-on and delegation but in this paper, these have been implemented on Alchemi.NET based grid computing framework through X.509 proxy certificates.
Keywords: Alchemi .NET based Grid Computing Framework, X.509 Proxy Certificates, Grid Security Infrastructure

1. Introduction

Grid computing usually consists of one main computer that distributes information and tasks to a group of networked computers to accomplish a common goal. Grid computing in one way deals with applying the resources of many computers in a network to a single problem at the same time [1]. Usually the problem taken is a scientific or a technical problem that requires a large number of computer processing cycles or access to large amounts of data. A task that involves multiple computers, multiple parallel transactions or other events, is a good candidate to be run using the computing grid [2].

Software to enable grid computing has been primarily written for Unix-class operating systems, thus severely limiting the ability to effectively utilize the computing resources of the vast majority of desktop computers i.e. those running variants of the Microsoft Windows operating system. Addressing Windows-based grid computing is particularly important from the software industry's viewpoint where interest in grids is emerging rapidly [3]. However, there is a distinct lack of service oriented architecture-based grid computing software. To overcome this limitation, Windows-based grid computing framework called Alchemi has been designed [2] and implemented on the Microsoft.NET Platform. In this paper, single-sign-on and delegation mechanisms have been implemented in Alchemi .NET based grid computing framework through X.509 proxy certificates.

[1,2]Department of Computer Science & Engineering, Panjab University, Chandigarh, India

Email: [1]ermanisha.cse@gmail.com, [2]sarbjeet@pu.ac.in, [2]singhmakhan@rediffmail.com

2. Architecture of a Basic Desktop Grid Computing System

To develop grid applications, users utilize the API's [2] and tools to interact with a particular grid middleware. When they submit grid application for processing, units of work are submitted to a central controller component which coordinates and manages the execution of these work units on the worker nodes under its control. Figure 1 shows the architecture of a basic desktop grid computing system. There are a number of considerations that must be addressed for such a system to work effectively which are as follows [2]:

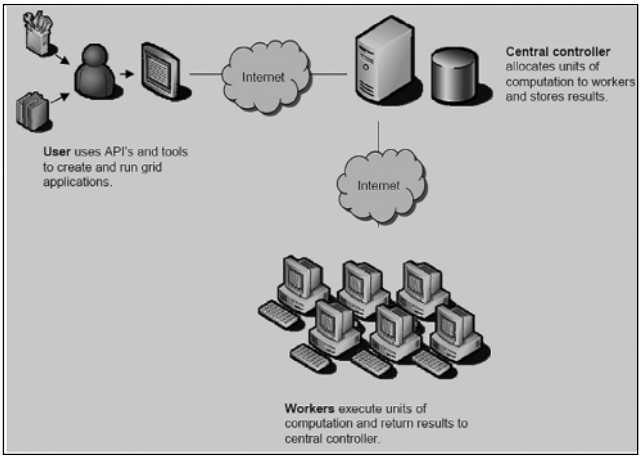

Fig. 1: Architecture of Basic Desktop Grid [2]

(a) Security Barrier - Resource Connectivity behind Firewalls: Firstly, worker nodes and user nodes must be able to connect to the central controller over the Internet and the presence of firewalls and/or NAT servers must not affect the deployment of a desktop grid [3].

(b) Unobtrusiveness - No Impact on Running User Applications: The execution of grid applications by worker nodes must not affect running user programs [3].

(c) Programmability - Computationally Intensive Independent Work Units: As desktop grid systems span across high latency of the Internet environment, applications with a high ratio of computation to communication time are suitable for deployment [3].

(d) Reliability – Failure Management: The unreliable nature of Internet connections also means that such systems must be able to tolerate connectivity disruption or faults and recover from them gracefully. In addition, data loss must be minimized in the event of a system crash or failure [3].

(e) Scalability – Handle Large Users and Participants: Desktop grid systems must be designed to support the participation of anonymous or approved contributors ranging from hundreds to millions. In addition, the system must support a number of simultaneous users and their applications [3].

(f) Security – Protect both Contributors and Consumers: Finally, the Internet is an insecure environment and strict security measures are imperative. Specifically, users and their programs must only be able to perform authorized activities on the grid resources. In addition, users/consumers must be safeguarded against malicious attacks or worker nodes [3].

3. ALCHEMI LAYERED ARCHITECTURE

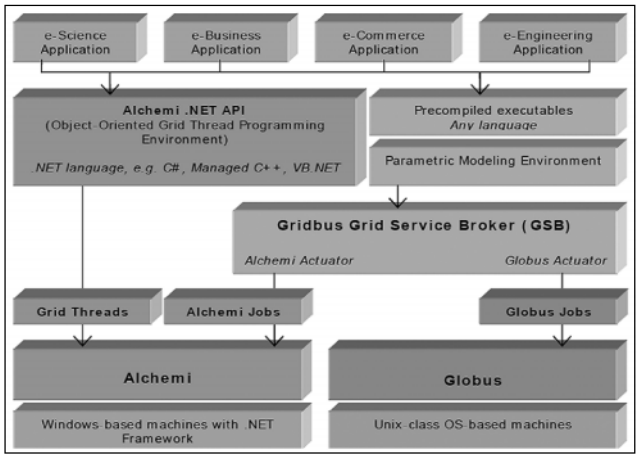

Fig. 2: Layered Architecture for Integration of Distributed Windows and Unix-class Resources [3]

A layered architecture-based grid computing environment using Alchemi and other grid technologies such as Globus Toolkit is shown in Figure 2. Alchemi follows the master-worker parallel computing paradigm in which a central component dispatches independent units of parallel execution to workers and manages them [3]. In Alchemi, this unit of parallel execution is termed 'grid thread' and contains the instructions to be executed on a grid node, while the central component is termed 'Manager'. A 'grid application' consists of a number of related grid threads. Grid applications and grid threads are exposed to the application developer as .NET classes / objects via the Alchemi .NET API. When an application written using this API is executed, grid thread objects are submitted to the Alchemi Manager for execution by the grid [2].

4. PROPOSED WORK AND IMPLEMENTATION

Alchemi is a .NET-based Grid Computing Framework and it does not support mechanisms like Single Sign On and Delegation which are present in My Proxy Credential Repository. So it is desired to add these mechanisms into Alchemi to increase its usability and applicability.

Alchemi is an open source software framework that has been specifically designed for the fast and efficient harnessing of the computational power of networked machines. The Alchemi framework is written in the C# programming language [2]. The Alchemi software is designed for application level task definition which is tied into applications through the use of the object oriented programming concept inheritance. The interface provided is extremely simple to utilize in any existing C# application. The architecture is such that jobs are sent to a manager machine whose responsibility it is to handle scheduling and manage the available resources on the grid. Client machines, known as executors, connect to the manager at will and in doing so make themselves available for use by the manager for executing jobs [3]. By default, the manager makes scheduling decisions based on the availability of CPU resources of the executor machines.

The manager is also responsible for handling exceptional circumstances such as when executor machines unexpectedly disconnect either on purpose or due to network failure. Microsoft .NET Remoting is the transport technology utilized by Alchemi to remotely execute code on machines in the grid. Security plays a key role in an insecure environment such as the Internet. Two aspects of security addressed by Alchemi are: (a) allow users to perform authorized operations whether they are system related or resource related operations and (b) allow authorized or non-authorized users to contribute resources. Alchemi .NET based Grid framework does not support Single sign on & delegation mechanism which are being implemented by the Myproxy Credential Repository [7], [8] using Proxy certificates [4] so there is need to implement these mechanisms in Alchemi.

5. PROPOSED SINGLE SIGN-ON ALGORITHM

1. A trusted communication is created between host A and host B.
2. User request host B to create a proxy.
3. Host B creates the request for the proxy certificate, and sends it back to host A.
4. Host A signs the request to create user's proxy certificate using user's private key and sends it back to host B.
5. Host A sends the certificate to host B.
6. A user or application on A sends its certificate to the host B.
7. Host B will get the public key for A and will use it to extract the subject from the certificate.
8. Host B creates a unique random number and sends it to host A.
9. Host A receives the number, encrypts it with its private key, and then sends the encrypted number to host B.
10. Host B will decrypt the number and check that the decrypted number is really the one that it sent to host A. Then host B authenticates that the certificate is really that from the user on host A, because only that user on host A can encrypt the number with its private key.
11. The certificate is authenticated by host B and user is allowed to access Alchemi .Net based grid computing framework and all its application with single sign on and server can delegate its responsibilities to the user whenever required.

According to this algorithm, firstly client has to create X.509 proxy certificate, which is sent to the server side and then server will extract the subject from the proxy certificate. After retrieving all information from the proxy certificate, server will generate a unique random number and send it to the client then client will encrypt that very random number with its private key and send it back to the server side which further decrypted by the server. Finally if the random number which was sent before is same as the decrypted random number then client is authenticated to use the Alchemi computational grid. Thereafter, server provides different roles to clients and delegate the responsibilities to them as per requirement. Therefore in this way user is able to use various applications available within the Alchemi .NET based grid computing framework with single sign-on. Single sign-on allows a user to provide his or her credentials once in order to access multiple applications.

6. DEMONSTRATION THROUGH A CASE STUDY

We have chosen two domains one of which acts as the client side & the other acts as the server side. For creating the application for Alchemi .NET-based grid computing framework, we have used C# as programming language & MS SQL Server as the database. When client needs to access the application like E-business then client has to go through the various steps as mentioned in the above proposed single sign-on algorithm.

When the user is authenticated then user is allowed to access the applications available in Alchemi .NET-based grid computing framework with single sign-on. After authentication, client is allowed to choose any of the application as shown in following Figure 3. In this demonstration, we have taken E-business application in which client is allowed to select a particular category of the products available in the E-business market.

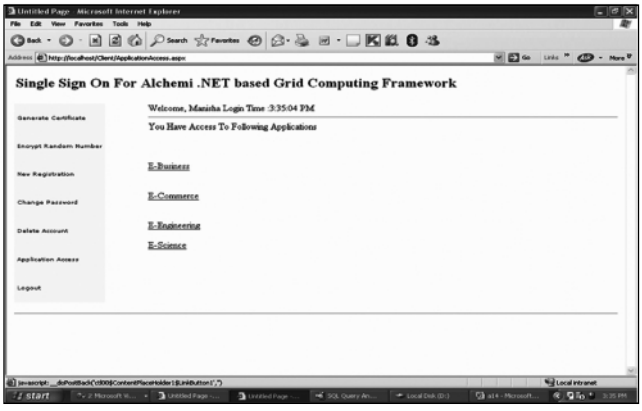

Fig. 3: Applications Available in Alchemi

Once the user logouts after accessing the applications with single sign-on, all information of the user is removed from the database like his certificateId, subject, public key etc. Alchemi with single sign-on and delegation mechanisms provide user an easy access to various applications by logging to the Alchemi system only once.

7. CONCLUSION

Advances in networking technology and computational infrastructure make it possible to construct large-scale high performance [9] distributed computing environments that provide dependable, consistent and pervasive access to high end computational and heterogeneous resources despite geographical distribution of both resources and users. In a heterogeneous computing environment like Grid, a suite of different machines ranging from personal computer to supercomputer is loosely inter-connected to provide a variety of computational capabilities to execute collections of application tasks that have diverse requirements. These kinds of large scale high performance distributed services have

recently received substantial interest from both research as well as industrial point of view. However, an important research problem for such systems is the lack of single sign-on and delegation system. Many kinds of credentials may be required for accessing a computational grid. After studying the PKI – based authentication mechanisms [9] in grid system and authorization models for proxy certificates in different grid middlewares, we have chosen to integrate single sign-on and delegation mechanisms in Alchemi .NET-based Computational grid. After analyzing the detailed performance of Alchemi, we have identified that Alchemi .NET-based Grid Computing Framework does not support mechanisms like single sign-on and delegation which are present in MyProxy Credential Repository [6] so it is desired to add these mechanisms into Alchemi to increase its usability and applicability. So we have implemented the single sign-on & delegation mechanisms for Alchemi .Net-based grid computing framework by making use of X.509 proxy certificates. In this implementation a trusted communication has been set up between two hosts. Then the host who is acting as the client have been authenticated by the host acting as the server. After authentication, Client is allowed to access various applications provided by the Alchemi .NET-based grid computing framework with single sign-on and full delegation rights.

REFERENCES

[1] I.Foster, C.Kesselman and S.Tuecke, "The Anatomy of the Grid:Enabling Scalable Virtual Organizations" International Journal of High Performance Computing Applications, Pg 200-222, 2001.

[2] A. Luther, R.Buyya, R. Ranjan, and S.Venugopal "Alchemi: A .NET based Grid Computing Framework and its Integration into Global Grids" Grid Computing and Distributed Systems (GRIDS) Laboratory Department of Computer Science and Software Engineering, The University of Melbourne, Australia, 2004.

[3] A. Luther, R. Buyya, R.Ranjan, and S. Venugopal, "Peer-to Peer Grid Computing and a .NET-based Alchemi Framework" Grid Computing and Distributed Systems (GRIDS) Laboratory Department of Computer Science and Software Engineering The University of Melbourne, Australia, 2005.

[4] S. Tuecke, V.Welch, D. Engert, L. Perlman, and M. Thompson,. RFC3820: "Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile", 2004.

[5] V. Welch, I. Foster, C. Kesselman, O. Mulmo, L. Pearlman, S. Tuecke, J.Gawor, S. Meder, and F. Siebenlist, "X.509 Proxy Certificates for Dynamic Delegation" In 3rd Annual PKI R&D Workshop, 2004.

[6] J. Basney, M. Humphery and V.Welch, "The Myproxy Online Credential Repository", Software: Practice and Experience, 35, Issue 9, pp. 801-816., 2005.

[7] J.Novotny, S.Tuecke & V.Welch, "An Online Credential Repository for the Grid: MyProxy", available at http://www.globus.org

[8] J.Basney, W.Yurcik, R.Bonilla and A.Slagell, "The Credential Wallet: A Classification of Credential Repositories Highlighting MyProxy," 31st Research Conference on Communication, Information and Internet Policy (TPRC 2003), Arlington, Virginia, September 19-21, 2003.

[9] S.Zhao, A.Aggarwal and D. Kent, "PKI-Based Authentication Mechanisms in Grid Systems" School of Computer Science University of Windsor, Canada, 2003.