Volume 4, Issue 7, July 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Improving Data Security and Efficiency in Grid Computing using Object Based Grid Architecture

Omerah Yousuf *
Department of Computer Science and Engineering, AMC Engineering College, Bangalore, India

Ab Rouf Khan
M. Tech, SCSE
VIT University Vellore, Tamilnadu, India

Vairamuthu S
Sr. Assistant Professor, SCSE
VIT University Vellore, Tamilnadu, India

Abstract – With the advent in the recent technological trends and the mammoth growth of worldwide networks, the distributed elaboration systems have become more and more powerful in terms of both the computing as well as the storage capabilities. In grid computing, the data is distributed among several nodes and is shared logically among several users. As the data is shared among the different users, there arises the need for the 'security' in grid computing, which essentially incorporates the issues like a) confidentiality, b) authorized users availability and c) integrity of the information and data to be shared.

In this paper we are proposing architecture for improving data security and efficiency in the grid environment based on the object oriented concepts. The main focus is given to enhance the security of the 'data' and the 'processes' which operate on these data, which essentially are both public in nature as far as grid computing is considered. Using the object based architecture the privacy related to the 'data' and the 'process' can be increased, thereby improving the Security of the overall system. The efficiency can be improved by the efficient use of the efficient threading concept in parallel architecture.

Keywords *– Grid Computing, Data Security, Confidentiality, Object Oriented Design, Parallel Architecture, Threading.*

I. INTRODUCTION

We belong to a generation which has seen leap-frog developments in the technology in the recent past. Be it the modern era smart phones or the next generation extensive computing techniques, the life in the future is going to be ruled by the next generation technologies. As the technology behind the development of these advanced computing techniques became more efficient, people demanded the more and more from these. One such computing technique which is going to rule the next era is 'Grid Computing'. It is because of the fact that the next generation computations will require a huge amount of resources and a very more processing power. Be it a collaboration of a group of civil engineers to design, analyse and execute the huge amounts of data or be it visualisation, annotation, and analysis of terabyte simulation datasets by scientists working in Meteorological department, all require the collaboration of huge amount of resources and much higher processing power than a single processor [1]. Thus, there arises a need of an efficient technique which can help us in resolving these issues. And this reason gave birth to a new computational technology known as Grid Computing.

Grid computing is the collection of computer resources from multiple locations to reach a common goal. The grid can be thought of as a distributed system with non-interactive workloads that involve a large number of files [2]. Grids tend to be more loosely coupled, heterogeneous, and geographically dispersed [3]. 'The Grid' takes its name from an analogy with the electrical 'power grid'. The idea was that accessing computer power from a computer grid would be as simple as accessing electrical power from an electrical grid" [3]. In the modern day world there are millions and billions of computing devices owned by a huge number of people or organisations or even IT firms. These different computing devices range from a tiny mobile phone to a huge supercomputer. The processing powers of all these devices vary a lot from each other. If we can combine these varying processing power devices into a common platform using the efficient networking techniques, we can be able to generate a collaborative system capable of performing any task. Means we can achieve a hell lot of both processing power as well as the resource sharing. This will eventually lead to a formation of system which can address all our issues as discussed previously.

There are a number of other scenarios too where we need an extensive computing technique like Grid Computing which can be summarised as:

- Imagine a scientist studying the Genomics of the human race and need to arrive at some conclusion based on his number of parameters of research [1],
- An IT firm accessing a huge chunk of databases of stocks in order to predict their future in a particular stock,
- An Army application in a civil war or generating the data related to the security of country,
- Real time response of an Emergency team in a situation of catastrophes or any other natural disasters [1],
- Analysing the huge amount of data available related to the different aspects of physical world.

All the scenarios discussed above require an efficient computing technique which can not only process the data but also do it well within the deadline or a given time frame. Basically we can think of grid computing as an architecture consisting of a computer network with each and every computer sharing its '**all**' of the resources, processing power, memory (physical), data storage components (external), external peripherals within an authorised network. We must keep one thing in mind and that is a grid computing system can be a small collection of computers running on a similar type of operating system or a huge and complex inter-networked system involving devices ranging from a simple personal computer to a most powerful super computer.

Some authors refer grid computing as a special case of Distributed computing. However the main difference between the two lies in the fact that in a Distributed System architecture and a Grid computing one lie in the fact that in Distributed system architecture *one or more* resources are shared by computers on a same computer network while in a Grid Computing architecture *all* the resources are shared ideally. We can also express this view in some another way in which we can reach a conclusion that a Grid is essentially a hardware and software infrastructure that clusters and integrates high end computers. The different networks, databases and other instruments from multiple sources form a type of super computer in which users can work in collaboration within virtual organisations. In another perspective we can define Grid Computing as: 'An architecture which uses physical hardware to do computations as if it was a single hardware'. Also we must note that there are a lot of differences between cloud and Grid computing [5]. A cloud could usually use a grid; However, a Grid is not necessary a cloud or a part of cloud.

Not all the technologies come without some drawbacks. The key challenges to the mobile cloud computing are:

1. Network availability, availability of resources and processing power, Scheduling, Data Management.

2. Security

    - Authentication
    - Authorization

This paper however aims at the later disadvantage. How to provide the security regarding the personal information about the users and the authenticity and integrity of the data they use and share is one the biggest challenges posed to the Grid computing in the recent past. Recent surveys by number of internet security developers reveal that there has been a sharp rise in the malwares since last quarter of year 2010. This makes the security to become more and more important, as the people can't afford to keep themselves away from this type of technology which can address a vast number of issues of the present era. But that does not mean it comes at the cost of security of very important data. So we need to think of a mechanism which can provide security to the data of users, while enjoying the services of this extensive computing technique.

II. LITERATURE SURVEY

There has been a lot of research done in the area of Grid Computing security issues carried out by the prominent authors in the recent past. Since it is one of the hot research areas of the 21st century, we can easily find a huge amount of research material on which this paper can be built and the new mechanism which will be efficient than the existing approaches will be proposed. In year 2006, Ching Lin, Vijay Varadharajan, Yan Wang and Vineet Pruthi, in their paper titled '*Enhancing Grid Security with Trust Management* [6]*'* discussed how grid security can be improved with the help of Trust Management architecture capable of capturing various types of f trust relationships that exist in a Grid system and providing mechanisms for trust evaluation, recommendations and update for trust decisions. And the outcomes of these trust decisions were used to enforce the different trust enhancement security solutions. Again in the year 2006, Matthew Smith, Michael Engel, Thomas Friese, Bernd Freisleben, in the paper titled '*Security Issues in On-Demand Grid and Cluster Computing'* [7] discussed the different issues related to security related to on-demand Grid were addressed. The analysed the different aspects of the security related to authentication, authorization, secure communication and confidentiality were discussed. They developed a security mechanism model called as: Trusted Computing Platform Alliance (TCPA). The presented security mechanisms increase the resilience of the service hosting environment against both malicious attacks and erroneous code. In year 2008, during the proceedings of Fourth IEEE International conference on eScience, Yuri Demchenko, Cees de Laat, Oscar Koeroo and David Groep, in the paper titled '*Re-thinking Grid Security Architecture'* [8] aimed at sharing of some observations on current security models and solutions found in Grid architectures and deployments today and identify architectural limitations in solving complex access control and policy enforcement scenarios in distributed resource management. The paper clearly provides a deep information related to overview of the OGSA security services and other security solutions used in Grid middleware and operations practice. However, it is becoming clear that further development in Grid requires a fresh look at the concepts, both operationally and security wise. In year 2009, during the proceedings of Fifth International Conference on Information Assurance and Security, Li Mingming, Li Baiping and Li Wei, in the paper titled '*Information Security Wireless Sensor Grid'* [9] analyzed the necessity and key problems of combining the WSN and grid computing and shows a sensor grid architecture which is based on an embedded wireless gateway. Security strategy of this architecture which is based on AES data encryption standard, digital signature and RSA Public key encryption algorithm is proposed in the said paper. In the year 2010, Guohua Liu, in the paper titled, '*Using Security Proxy based Trusted Computing*

Enhanced Grid Security Infrastructure' [10] discussed the concept of Grid Security and addressed the different issues related to it. With the help of existing Grid security solutions like Grid Security infrastructure (GSI) for Globus Toolkit (GT), they developed a system called as Security Proxy based Trusted Computing (SPTC) modular to enhance the grid security. In the year 2011, during the proceedings of 2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN 2011), M Victor Jose and V.Seenivasagam, in the paper titled *'Object Based Grid Architecture for Enhancing Security in Grid Computing'* [11] described a simple logical method to enhance the security in Grid Computing. They proposed an Object Based Grid Architecture (OGA) to enhance the security of Grid Computing. The architecture was based on object oriented concepts. In OGA the data and process privacy is increased through an Object Oriented Grid Platform and with many other features. In the year 2012, N.Sandeep Chaitanya, S.Ramachandram, K. RamaKrishna, S. Siva Skandha and B.Padmavathi, in the paper titled *'RAID Technology for Secured Grid Computing Environments'* [12] discussed the challenges and the existing solutions for Grid Computing. In the paper the authors presented a new method of storing electronic records distributed at different systems site using RAID-3 algorithm. By RAID-3, the segmented data stored at each system is meaningless and cannot be used alone. This will enhance the security & Privacy of data.

III. PROPOSED ARCHITECTURE FOR DATA SECURITY IN GRID COMPUTING USING OBJECT BASED GRID ARCHITECTURE

The architecture proposed in the paper has two fold advantage, 1. It aids in imparting Security in Grid Computing Architecture & 2. It can improve the performance of Grid Computing architecture on a whole. The architecture has been design mainly to extract the different features of Object Oriented methodology to improve the security as well as efficiency of the grid computing. The designed architecture mainly consists of different objects. The objects themselves contain the different clusters which possess different nodes used to perform the different computations. All the processing is done at the nodes in the clusters. The nodes interact with each other in a same cluster and the different clusters in a single object communicate with each other in a bi-directional way.

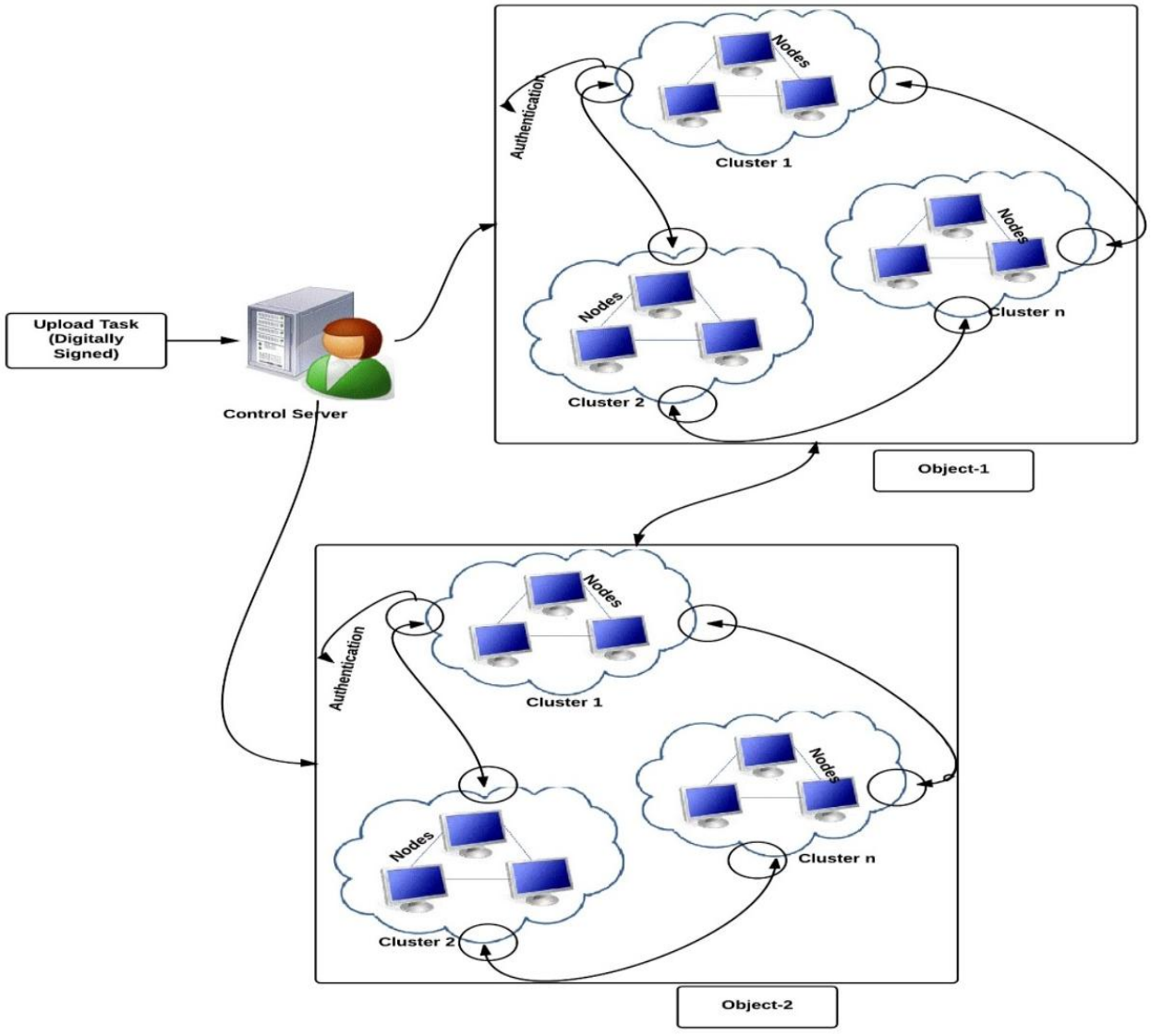

Figure 1: Proposed Architecture to improve Data Security and Efficiency of Grid Computing

Fig1: Architecture Proposed to Improve the Data Security and Efficiency of Grid Computing

Fig 1 shows the proposed architecture to improve the Data Security and Efficiency of Grid Computing. The architecture consists mainly of *objects*. There can be as many number of objects, as the problem at hand demands. In the design above only two objects have been shown in order to make the visualisation better. The objects can communicate with each other whenever they need to exchange data or any other resource. Before going in deep about the designed architecture we must be able to understand why there arises the need for security in the Grid Computing? It is mainly because of four factors [1] [2]:

- Resources are Distributed
- There is no Centralized Control on the system
- The Resources are provided by different resource Providers (Vendors)
- Each and every Resource provider may use different security policy at its level of operating the resources and system.

Considering the factors mentioned above we can reach to a conclusion that Security is one of the major challenges in the Grid Computing architecture, which needs to be addressed very wisely. Let us consider the two broader aspects of the proposed architecture i.e., Improving Data Security and Improving Efficiency separately as follows:

1. Improving Data Security

The architecture can be explained in the view of data Security and Efficiency by taking a scenario as: Consider a scenario in which a biochemist needs to exploit a hundreds and thousands of computers to analyse the synthesis of proteins in humans. Since there are a number of steps involved in the synthesis of the proteins, there can be a number of steps that work independently of each other. The biochemist will upload the digitally signed task to be processed using the 100's of computers connected together on web.

The digitally signed task to be processed is verified for the security by the Control server and once it is confirmed that the file is authentic and does not contain any malicious thing that can affect the security of the system, the file is eligible for the further processing now. The user (biochemist in this case) will be sent an acknowledgement about the file being accepted as a genuine one only if the user has opted for the same. Once the file is accepted as a genuine file free from malicious things, the control server decides what and how the processing of the file should take place. Let us explain the sequence of flow of events in view of the architecture proposed where the system is supposed to consist of only two objects. The control server will distribute the given task among the objects in a best possible way so that the independent sub tasks are supplied to the two objects and both the objects share the load equally or nearly equal.

Once the task is divided in such a way that the independent sub tasks are mounted on the two objects to be processed, the independent sub tasks will be processed parallel by the different clusters in a particular object. Since the objects will be possessing the default ability to encapsulate the data which is one of the primary characteristics of the object oriented architecture, it will help in securing the data as a whole in the system from a broader perspective. Once the data enters the particular object to be processed on the nodes, the nodes being located inside a cluster the data needs to follow and accept the security protocols that will be enforced by these clusters. At the entry point of each cluster we are using an *authentication* point. At the authentication point the data is checked again whether it has been changed while passing through the different communications channels into the object. The file is verified against the task digitally signed by the user. Once the file is found free from the malicious things, the actual processing starts. Each of the clusters in an object follow the authentication process once the data enters the cluster. It is worth to mention here that any of the authentication mechanisms can be used to authorize the data, it can be cryptography or any of the algorithms used in the secure clustered networks. The architecture proposed here is independent of the authentication principle used. The architecture proposed just provides a frame work in the way we can impose the security constraints in the Grid Environment. The authentication is to be followed at each and every point whenever the data enters a particular cluster. It must be noted that the actual processing of the data occurs only at the nodes in a cluster. One can argue that, in case the two objects need to communicate with each other, why not the authentication process is employed at this point. It is because of the fact that we assume by default, that the objects are encapsulated and there is no chance of the data impairment.

Other than the authorization points or nodes at the entry of the each cluster, the data inside the clusters after being processed needs to be secured also. At the cluster level, we will employ any of the cluster data security algorithm like *Distributed Cluster (Clique) Algorithm* [11]. So the data inside the clusters is secured as well.

As per the perspectives of Data Security we can argue that the proposed architecture provides a three-tier security to the data:

1. Encapsulation of data in an object (Default Property of Object Oriented approach),
2. The authorization or authentication entry at each of the entries of a cluster, and
3. The security at the cluster level using any cluster data security algorithm.

Hence we can conclude that the proposed architecture can act as a framework to implement the Data Security concept in the Grid Computing architecture efficiently using the object oriented concept.

2. Improving the Efficiency

Efficiency can be improved in the Grid Computing architecture using the proposed architecture by the means of executing the particular task wisely enough in terms of parallel and serial execution. Once the task is uploaded by the user the control server will divide the given tasks into the independent tasks and based upon the objects used in the architecture which is eventually determined by the complexity of the task the number of threads will be created by the master thread controlled by the Control Server itself. Using the different threads we can execute the work in parallel. Now there arises the need how to assign the threads in order to get the better performance with the available resources. We will take the help of Open MP work sharing constructs [12]. Open MP is actually one of the Parallel Programming models. Among the different work sharing constructs provided by the Open MP model we will mainly use *parallel* and *for* work sharing constructs. The *parallel* work sharing construct helps us in executing the task in parallel. Control server will assign the number of threads required to execute the sub tasks. Inside the object we have different clusters having the different processing nodes where the actual processing takes place. The thread assigned to each object can be further made more efficient by performing the tasks performed by different clusters parallel too. This can be done using *for* construct of the Open MP model. The process flow can be explained with the help of figure 2 below.

```
#pragma omp parallel
{
    #pragma omp for
    for (i=0; i< MAX; i++) {
       res[i] = huge();
    }
}
```

Fig 2: Usage of Workflow Constructs in Open MP Model

We can control the behaviour of threads using the Schedule Clause of the Open MP model. When the task at hand is small and we need to perform the Predictable and similar work per iteration we will use the *static* scheduling clause. It is having less overhead and very high efficiency when the task at hand is small and there is similar type of work per iteration. And in case the task at hand is very big and we need to perform very high number of computations, we will use the *dynamic* scheduling clause of Open MP parallel programming model. The dynamic scheduling clause is very much useful when we have unpredictable, highly variable work per iteration.

Thus we can argue that by controlling the number of threads and the behaviour of threads in turn, we can increase the efficiency of the Grid Computing on a whole.

From the observations drawn based on the execution of the process of the protein Synthesis data in terms of Execution time taken to execute the complete process on different core machines, we can conclude that if the data size is small, the lesser core machines are better to be used than higher core machines in order to reduce the overhead time, as the computation time will not be longer in the case of small sized problem. Once the data size increases the higher core machines are highly efficient as compared to lesser core machines. This is because of the fact that we need a huge number of computations and the overhead time is very less as compared to the number of the computations required, and the higher core machines will perform the computations much faster than the lesser core machines. The observations performed on the variable size data and the variable number of core machines can be visualised properly with the help of table below (Table 1).

Table 1: Execution time analysis on different core machines

Data Size (No. of genes participating in protein synthesis)	2-Core parallel Execution time (sec)	4-core parallel Execution time (sec)	8-core parallel Execution time (sec)
5000	0.2193	0.1279	0.1050
10000	1.1254	0.4396	0.3900
15000	2.7595	1.2476	1.1200
30000	9.5391	7.0260	3.9250
40000	22.208	13.4302	8.4632

IV. CONCLUSION

Security is one of the major challenges in the Grid Computing architecture, which needs to be addressed very wisely. In the paper a frame work is proposed to improve the Data Security and Efficiency of the Grid Computing using the Object based architecture. The proposed architecture provides a three tier security to the data – Encapsulation at the object level, Authentication at the Entry points to the clusters and Authorization and authentication inside the clusters. The Efficiency can be improved with the help of controlling the number of threads used to execute a particular task and controlling the behaviour of the threads while they are executing. It can be concluded that the proposed framework in this paper can aid in improving the Data Security as well as Efficiency in Grid Computing Environment.

REFERENCES

[1] Ian Foster "The Grid: A new infrastructure for 21st Century"
[2] Ian Foster "What is the Grid? A Three Point Checklist", Year 2002.
[3] HowStuffWorks Inc., "How Grid Computing Works", © 1998-2014 HowStuffWorks, Inc.
[4] Ian Foster, Yong Zhao, Ioan Raicu and Shiyong Lu, "Cloud Computing and Grid Computing 360-Degree Compared", Year 2002.
[5] Ioan Raicu, "Cloud Computing and Grid Computing 360-Degree Compared", IEEE Fox Valley South Section, IIT January 14th 2011.
[6] Ching Lin, Vijay Varadharajan, Yan Wang and Vineet Pruthi "Enhancing Grid Security with Trust Management", Proceedings of the 2004 IEEE International Conference on Services Computing (SCC'04)
[7] Matthew Smith, Michael Engel, Thomas Friese, Bernd Freisleben, Gregory A. Koenig, and William Yurcik, "Security Issues in On-Demand Grid and Cluster Computing", Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid Workshops (CCGRIDW'06).
[8] Yuri Demchenko, Cees de Laat, Oscar Koeroo and David Groep, "Re-thinking Grid Security Architecture", Proceedings of Fourth IEEE International conference on eScience – 2008.
[9] Li Mingming, Li Baiping and Li Wei, "Information Security Wireless Sensor Grid", Proceedings of Fifth International Conference on Information Assurance and Security – 2009
[10] Guohua Liu, "Using Security Proxy based Trusted Computing Enhanced Grid Security Infrastructure", IEEE 2010.
[11] M Victor Jose and V.Seenivasagam, "Object Based Grid Architecture for Enhancing Security in Grid Computing", the Proceedings of 2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN 2011)
[12] N.Sandeep Chaitanya, S.Ramachandram, K. RamaKrishna, S. Siva Skandha and B.Padmavathi, "RAID Technology for Secured Grid Computing Environments", IEEE – 2012

