Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 1, Number 3; November, 2014 pp. 131-134
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/acsit.html

Security Implementation in Parallel Program Execution of Distributed Systems

Ms. Prabhjot Kaur

Guru Nanak Institute of Management & IT Punjabi Bagh, New Delhi
E-mail: ginni5.gold@gmail.com

Abstract: *The paper is dealing with the comparative study on the security implementation for the parallel running programs of distributed system. We know that distributed system involves working of many systems simultaneously which is diversely located but act as a single system. So, various security issues are involved in handling this type of complicated systems. This paper discusses the security issues associated with various types of distributed systems present. Based on the analysis of the security issues faced by the three types of distributed systems, solution is discussed for a system which executes the program in parallel. A system is discussed which is using client–server architecture for implementing distributed system for parallel evaluation of the jobs. This system is easy to use, easy to compute and easy to understand which has the centralized concept of managing the applications. In this system, all the applications are centrally located as the repository of all the applications being worked at diverse locations of a single acted system. Thus central management of applications is needed on client side for executing the job which involves security measures. Network File Protocol (NFP) is used in this purpose. Through this system we can target the small scale project, which runs over a LAN. The system is meant for solving problems which can be fragmented but in expects no real communication between clients. Due to lack of interconnection between clients, it is required that jobs should be self contained so as to independently process assigned jobs. In such kind of working atmosphere, various security measures are studied, under which a new security approach is resulted.*

Keywords: *Security; Distributed System; Parallel execution of programs; small scale projects*

1. INTRODUCTION

In today's world, computers cannot work in isolation. A network is required so as to communicate, process, transfer data etc. When systems work in a networked environment, then the environment involved is geographically diverse in nature. These diversely located systems which together work as a single system is known as distributed system. There are many definitions for distributed systems such as "A distributed system is a collection of hosts interconnected by a network", "A collection of systems that appears to the users as a single system". But Tanenbaum's definition best stated that a distributed system is a collection of independent computers that appears to the users of the system as a single coherent system.

In this type of the system, autonomous computers are linked by a computer network that appears to the users of the system as a single computer. All the computers in the system work independently. It is perceived as a single system to solve a certain problem. All the systems in the distributed network share resources like hardware, software, data etc.
Distributed system was built with the following objectives:-

- Reliability
- Transperacy
- Openess
- Performance
- Scalability

2. DISTRIBUTED SYSTEMS & ITS THREATS

There are various types of distributed systems. The most popular distributed systems are:-

- Clusters
- Grids
- Distributed Storage Systems

2.1. Clusters

A cluster is a dedicated group of interconnected computers that appears as a single super computer, generally used in high performance scientific engineering & business applications. Any task which is assigned in clusters would run in parallel on all the computers by breaking a large problem into smaller tasks which are solved independently on each system separately on a cluster. It helps the large scale organizations to increase their computing power.

2.2. Grids

A grid enables coordinated sharing and aggregation of distributed, autonomous, and heterogeneous resources based on user's Quality of Service (QoS) requirements. In this large number of small loosely coupled computers are brought together to form a large virtual supercomputer. In grids, with the help of middleware, we are able to divide a large problem into various small tasks which are distributed to the systems on the grid.

2.3. Distributed Storage Systems

Due to the growth in the storage capacity, bandwidth and computation resources, devices with reduced cost are required for storage purpose. Various distributed storage systems such as RAID, NAS, and SAN are popular.

Various threats are involved in distributed systems. Types of threats are:-

- Host: It is subversion of individual hosts in a system. It can be countered by a combination of hardware techniques and software techniques.
- Communication: It is the threat associated with message communications.

3. DESCRIPTION OF THE SYSTEM

We are discussing a system which uses client server architecture to perform the parallel execution of the task or programs efficiently. For this, the system has 3 main components:

- Web Interface at Server Side
- Client side
- Repository of data and resources

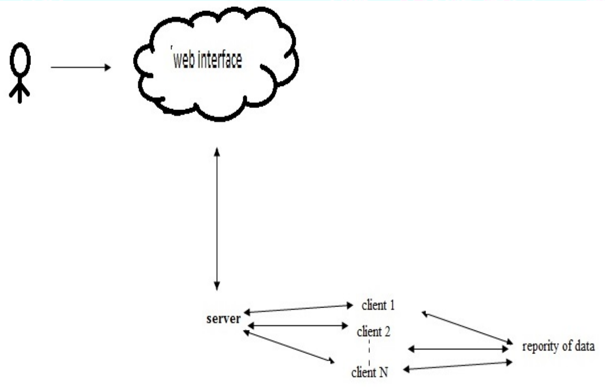

Fig. 1

As seen in Fig. 1, users submit their jobs via web interface to be accessed by the system. For each job user also defines the time limit for which the job is expected to get finished which is known as Time to Live (TTL). Also, user defines the memory required to execute the job. The TTL is used to get rid from those jobs which will never end or are stuck in infinite loop. User can even increase its Time to Live if he encounters that the job will not finish in the expected time limit give. User can even terminate any of its jobs at any point of time.

When user submits its job, they are classified into three categories: execution state, finished state or waiting to execute state. Monitoring of these jobs will result in updating the data of client side so as to view the job's completion level.

Once the client made itself available to the system, its current status of available resources is being periodically sent and maintained on the server side, based on which server chooses the most appropriate client for the new job.

When the job is assigned to the client, then all the required application is loaded from repository of data/resources and its computation time starts. User monitors all the computation done by the clients and remaining time to live is evaluated by them. When the job assigned finishes up then the entire record of the job is sent to the server which passes it to the web interface.

4. AUTHENTICATION

Authentication is basically identification along with the verification. Identification is basically a procedure in which an entity is claimed for a certain identity and Verification is a procedure that claims about the entity is checked. Authentication is mainly of two types: mutual authentication and one way authentication. Mutual authentication is the one in which both communicating principals verify each other's identity whereas one way authentication is the one where only one principal verifies the identity of the other principal.

There are major types of authentication exchanges in a distributed system:

- When there is an exchange of data within the hosts so they often require cooperation between the hosts.
- There is an authentication exchange between host and user too. As in an open access multiple hosts are scattered across area and user gains access to distributed system by logging in the host of the system.
- If a client is willing to surrender valuable information to a server then authentication exchange is required for server's identity verification.

While accounting for the authentication, there are many interests of authentication in distributed systems. The main three types are:

- Message content authentication in which all the data which is required to be sent from one place to another on a network, is verified by checking its contents on both the sides and it should be the way it is sent.
- Message origin authentication in which it is verified that the sender is the one whose address should be on the sender field of the message being sent.
- Identity authentication is the one in which it is verified that the data or the message which is sent should be identical.

5. WORKING OF THE SYSTEM

5.1. Server

A Server plays the main role in distributed systems. It has two basic functions–monitoring and scheduling. Monitoring function holds the list of currently connected clients and information about any time of the job execution. Scheduling function maintains the list of all user jobs which are waiting to be executed or the jobs which are currently executing or the finished jobs. Scheduling will decide which client will get the job for execution and to receive the result back from the user.

Thus basic functions of the server are:

- List of clients connected
- List of resources connected.
- Scheduling the job i.e. to select the client for executing the job.
- Web interface is maintained from the user
- Controlling newly arrived jobs and putting in queue for execution.
- Aborting the required jobs from execution whose time to live is exceeded.
- Changing time to live for jobs
- Collecting results of finished jobs from the client.

In this list of the clients is maintained which contains the record of each client with its resources connected in the network. Also job queues are being maintained for jobs which are waiting, jobs which are currently running on the clients and the jobs which are already executed.

5.2. Client

Client's job is to receive user's job, running it and sending the results back to the server. All the applications which are required for executing jobs are located on the repository, which is shared by all the clients attached to the repository.

Main functions of clients are:

- Tell the server about its existence.
- Reporting the current state of each job given to the client.
- Processing server's command.
- Receiving jobs from the server.
- Execute the job and report it to the server.
- When time to live finishes, jobs are aborted.

Client needs to maintain only the record of current active jobs. All the jobs which are currently running at client side is maintained and once the job is finished to be executed then it's records gets deleted from the record and the feedback is returned to the server for its completion declaration

6. SECURITY MEASURES

While working on this client server architecture, we have encountered that this system needs to be protected against the hackers and various security requirements of the applications need to be taken care of.

While allocating resources, security constraints need to be put in. Xie developed resource allocation schemes named "Security Aware and Heterogeneity Aware Resource allocation for Parallel Jobs (SHARP)". We can use this scheme because it ensures that parallel applications executed on client side meets the security requirements while meeting deadline of the job execution. Denial of service (DOS) attack is common while running parallel jobs on the distributed system. In this attack the resource get bounded in such a manner that they are prevented from carrying out their operational job which makes the executing job to come to halt and time to live is aborted by the user. Thus a method that uses services chain to mitigate the effects on DOS attack.

As we have earlier discussed that all the clients are attached to a common repository of data and resources, thus there may occur at times that number of clients at any instance of time may differ from the clients attached to the same repository of data and resources at other instance of time. Thus, it may occur that an application divided into the jobs of the clients, there will be some clients who are attached to the repository previously but at this particular instance not assigned the job. But that client can use the resources and data as it is attached. So at this point we require security measures.

To solve this we can introduce "authentication mode" to it. This is one of the critical components to prevent external clients from randomly access internal repository and the resources will get protected from unauthorized users. In the illustration, we can assign a new work to the user that it will distribute the jobs to the clients and while distributing jobs those clients will get authenticated to use the repository for completion of that job. Other clients which may be in the network but are unauthorized will not be able to carry out illegal operations with repository. Once the jobs are executed and application is finished executing then the authentication model is set again. This way we can prohibit the illegal use of resources.

7. CONCLUSION

In this paper, we have discussed the objectives of distributed system and three of its commonly used types. We also have demonstrated the easy to use and understand implementation of distributed system for efficient execution of parallel programs. We have discussed the security issues faced by the same model and proposed a solution to it.

8. FUTURE WORK

As we have proposed a solution to the security threats involved in the parallel execution, there are many more threats which need to be taken into account when this system is practically executed. So our future scope would be to implement this client server parallel execution through a programming interface and show this result of security into them. Also the upcoming security threats would be solved when executed on the network.

REFERENCES

[1] RajkumarBuyya and SrikumarVenugopal, “Market Oriented Computing and Global Grids” ; John Wiley & Sons,2010, ch 1

[2] Andrew S Tanenbaum and Maarten van Steen, Distributed Systems : Principles and Paradigms, Peason Higher Education, 2007

[3] N. Truhar, K. Veselic, An efficient method of estimating the optimal dampers, SIAM Journal on Matrix Analysis and Applications, 2009

[4] L. Peterson, T. Anderson, D. Culler and T. Roscoe, A Blueprint for introducing disruptive Technology in the Internet, 2002

[5] http://www.partimage.org/

[6] http://www.searchdatamanagement.techtarget.com