IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. X (Mar-Apr. 2014), PP 28-31
www.iosrjournals.org

Distributed Computing Challenges

[1]Mr. BhushanTalekar, [2]Miss SonaliChaudhari, [3]Prof. VinayakShinde, [4]Prof .GayatriMasiwal

[1,2,3,4,5]*Computer Engineering, Mumbai University Mumbai India,*

Abstract: *In recent years, distributed computing systems has been widely explored in order to improve system performance in terms of scalability and reliability.A distributed computing system allows the user to share, select, and aggregate the distributed heterogeneous computational and storage resources, which are under the control of different sites or networks. A distributed system also provides solutions to the complex scientific or engineering problems, such as weather forecasting, medical diagnoses, and stock portfolio management. However, the distributed nature of the system also raises some serious challenges. This paper covers the major challenges of security& design.However, complete distributed nature of the system raises serious challenges in domains of security like Scheduling,Objective Function,Security and Trust and also in domain of design like Heterogeneity,Openness,Reliability and Fault Tolerance, etcIn this paper, we aim to identify the challenges of distributed systems.*

Keywords: *distributed computing, securityChallenges, design Challenges.*

I. Introduction

Security is one of the leading challenges in developing distributed computing, since the integration of different components in a distributed manner creates new security problems and issues. Service oriented architectures ,the Web, grid computing and virtualization are some of the examples of today's distributed computing. Distributed computingsecurity provides a holistic insight into the current security issues, processes, and solutions, and maps out future directions in the context of todays distributed systems. Security Challenges are more critical in distributed systems because of Open architecture,Need for communication across heterogeneous systems across communication links.Being distributed in nature ,distributed computing has to face various challenges.

II. Distributed Computing Securitychallenges

2.1 Scheduling

In centralized scheduling approach, all the system wide decision making is done by a central controller, butthe decentralized scheduler organization hassome limitations.Two categories of scheduling are non-coordinated scheduling and coordinated scheduling.[1]

2.1.1 Non-Coordinated Scheduling

In the non-coordinated scheduling scheme, application schedulers perform scheduling related activities independent of the other schedulers in the system. Non-coordinated or non-cooperative scheduling is performed by directly submitting jobs to the condor pools without taking into account their load and utilization status. [1]

2.1.2 Coordinated Scheduling

In the Coordinated scheduling scheme, schedulers negotiates resource conditions with the local site managers in the system, if not, with the other application level schedulers. Legion-Federation system coordinates scheduling decision with other sites in the distributed environment through job query mechanism.[1]

2.2 Objective Function

Distributed computing are dynamic in nature and their states can change within small interval of time. Therefore, we need scheduling and resource allocation policies that can adapt to these changing resource conditions. As a result, the participants including resource providers and resource consumers associate various objective functions with respect to resource allocation and scheduling processes.We can distinguish the objective functions into two categories.[1]

2.2.1 System Centric

In system centric mechanism, a distributed computing defines relatively simple objective functions. A system centric scheduler focuses on maximizing resource throughput on the provider side, while minimizing overall consumer's application completion time.

2.2.2 User Centric

User centric scheduling objective functions based on QoSparameters . These QoS parameters include profit, reputation, security or combination of all, whereas QoS parameters for users are cost, budget spent, response time or combination of all.

2.3 Security and Trust

Distributed computing raises serious challenges in the domains of security and trust management. Following security issues:

- o Preserve the privacy of participants.
- o Ensure authenticity of the participants.
- o Provide robust authorization.
- o Route messages securely between distributedservices.[2]

2.3.1 Privacy

Privacy is one of the main challenges of security in distributed computing. The privacy of the participants can be ensured through secret key-based symmetric cryptographic algorithms, such as 3DES, RC4, etc. These secret keys must be securely generated and distributedin the system.

2.3.2 Authentication

Authentication means only legal person or the authorized person can access the data.Authentication of participants can be achievedthrough Public Key Infrastructure (X.509 certificates),Kerberos (third party authentication), distributed trust, and SSH.

2.3.3 Authorization

Authorization deals with the verification of an action that a participant is allowed to undertake after a successful authentication.

2.3.4 Secure Message Routing

Implementing secure and trusted message routingin distributed computing environment requires solution tothe following problems:

- secure generation and assignment ofnodeIds.
- securely maintaining the integrity of routingtables.
- secure message transmission betweenpeers.

III. Distributed Computing Designchallenges

3.1 Heterogeneity

A distributed computing system allows the user to share, select, and aggregatethe distributed heterogeneous computational and storage resources, which are under the control of different sites or networks. Because of heterogeneous nature distributed computing system faces the problem of networks (protocols), operating systems (APIs) and hardwareprogramming languages (data structures, data types) implementation by different developers (lack of standards).Distributed applications are typically heterogeneous:[2]

- Different hardware: mainframes, workstations, PCs, servers, etc.;
- Different software: UNIX,MSWindows, IBM OS/2, Real-time OSs, etc.;
- Unconventional devices: teller machines, telephone switches, robots, manufacturing systems, etc.
- Diverse networks and protocols: Ethernet, FDDI, ATM, TCP/IP, Novell Netware, etc.

3.2 Openness

One of the important challenges of distributed computing is openness and flexibility openness. It means every service is equally accessible to every client (local or remote) and it is easy to implement, install and debug services also users can write and install their own services.[2]

3.3 Reliability and Fault Tolerance:

One of the main goals of building distributed computing is improvement of reliability.

Availability: If machines go down, the system should work with the reduced amount of resources. There should be a very small number of critical resources; critical resources are the resources which have to be up in order the distributed system to work.

Fault-tolerance: is a main issue related to reliability the system has to detect faults and act in a reasonable way:

- Mask the fault: continue to work with possibly reduced performance but without loss of data Information.

• Fail gracefully: react to the fault in a predictable way and possibly stop functionality for a short period, but without loss of data/information.

3.4 Security**:**

Security of information resources:

3.4.1 Confidentiality :

Means Protection against disclosure to unauthorized person

3.4.2 Integrity :

Deal with Protection against alteration and corruption

3.4.3 Availability:

Keep the resource accessible to user at any time.

3.5 Performance and Scalability:

3.5.1 Performance

Several factors are influencing the performance of a distributed system:
• The performance of individual workstations.
• The speed of the communication infrastructure.
• Flexibility in workload allocation: for example, idle processors (workstations) could be allocated automatically to a user's task.

3.5.2 Scalability

The system should remain efficient even with a significant increase in the number of users and resources connected:
- cost of adding resources should be reasonable;
- Performance loss with increased number of users and resources should be controlled. [3]

3.6 Transparency

Deal with how to achieve the single system image? How to "fool" everyone into thinking that the collection of machines is a "simple" computer? It listed as different forms like:

3.6.1 Access transparency

Local and remote resources are accessed using identical operations.

3.6.2 Location transparency

Users cannot tell where hardware and software resources (CPUs, files, data bases) are located; the name of the resource shouldn't encode the location of the resource.

3.6.3 Migration (mobility) transparency

Resources should be free to move from one location to another without having their names changed.

3.6.4 Replication transparency

The system is free to make additional copies of files and other resources (for purpose ofperformance and/or reliability), without the user's noticing it. Example: several copies of a file; at a certain request that copy is accessed which is the closest to the client.

3.6.5 Concurrency transparency

The users will not notice the existence of other users in the system (even if they access the same resources).

3.6.6 Failure transparency

Applications should be able to complete their task despite failures occurring in certain components of the system.

3.6.7 Performance transparency

Load variation should not lead to performance degradation. This could be achieved by automatic reconfiguration as response to changes of the load; it is difficult to achieve.[1]

IV. Conclusion

To develop any distributed computing system it faces various challenges related to its security as well as design. In this paper much of the work has been focused on various types of security challenges and design challenges.

Acknowledgements

Thanks to Prof.VinayakShinde&Prof.GayatriMasiwalfor providing excellent guidance, encouragement and inspiration throughout the paper work. Without their invaluable guidance, this work would never have been a successful one. Last but not least, we would like to thank our family and friends, who have been a source of encouragement and inspiration throughout the duration of the paper.

References

[1] Al-Sakib Khan Pathan International Islamic University Malaysia, Malaysia,MukaddimPathan Australian National University, Australia,Hae Young Lee Electronics and Telecommunications Research Institute, South Korea" Advancements in Distributed Computing and Internet Technologies: Trends and Issues"

[2] Krishna Nadiminti, Marcos Dias de Assunção, and RajkumarBuyya"Distributed Systems and Recent Innovations: Challenges and Benefits".LalanaKagal, Tim Finin and Anupam Joshi University of Maryland Baltimore County"

[3] LalanaKagal, Tim Finin and AnupamJoshiUniversity of Maryland Baltimore CountyMoving from Security to Distributed Trustin Ubiquitous Computing Environments"