International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 24 Issue 8 – MAY 2017.

A Distributed Handling Mobility Handover base LTE Advanced Networks

A.UMA MAHESWARI[#1] and R Asha[*2]

[#] *Department of Computer Science and Engineering, Mahendra Engineering College, Namakkal, India*

[*] *Department of Computer Science and Engineering, Mahendra Engineering College, Namakkal, India*

***Abstract*— In this paper we have simulated the hard hand over mechanism of IEEE 802.16j transparent mode networks using NCTUns Tool. We have used Adaptive modulation and coding scheme. The results show when the mobile station moves form base station to base station, it has to disconnect the original connection with BS before making a new connection with BS. And also it shows that the throughput gradually decrease when the mobile station moves to NLOS. We identify the authentication and key agreement, procedure for each of the possible cases. We formally model these scenarios and analyze their security, in the symbolic model, using the tool ProVerif. We find two scenarios that inherit a known false base station attack. We find an attack on the CMC message of another scenario. A long delay and large computational overhead may occur during handover or roaming. Therefore, the security research of group-based communication in the duration of handover or roaming will be further exploited in our future work which demonstrate that the transmission overhead of the whole authentication is considerably reduced.**

***Index Terms*— Privacy, mobility, handover, Long Term Evolution (LTE), 3rd Generation Partnership Project (3GPP)**

I. INTRODUCTION

With the development of mobile communication systems, numerous authentication and key agreement (AKA) protocols have been proposed. To improve the security weaknesses in Global System for Mobile Communications (GSM)[1], UMTS-AKA, which is based on GSM's successor Universal Mobile Telecommunications System (UMTS), was proposed at the network level[2]for authenticating 3G mobile subscribers. UMTS-AKA can negotiate security keys between a subscriber and the serving network and then achieve mutual authentication between the two parties. UMTS-AKA can also successfully defeat most of the vulnerabilities found in GSM systems and ensure a more secure telecommunication environment. Nevertheless, it is still vulnerable to some sophisticated attacks, such as redirection and man-in-the-middle attacks. Recently, a novel authentication protocol dedicated for Evolved Packet System (EPS) has been proposed in the Long Term Evolution (LTE) project [3] by the 3rd Generation Partnership Project (3GPP), known as EPSAKA[4], which is based on its predecessor UMTS-AKA protocol. Backward compatibility of EPS-AKA is an important factor for its wide acceptance, but it may also hinder progress and limit the design freedom.

Our security analysis addresses authentication properties and secrecy. We show that two of the LTE interoperation scenarios inherit an attack, known from GSM and the interoperation between UMTS and GSM [6], in which a false base station can eavesdrop and modify data traffic. We show how the attack can be prevented in one scenario. We also show that one scenario is prone to an attack against the Cipher Mode Command (CMC) message.

II. BACKGROUND:

There has been many methodologies been proposed earlier to schedule the data oriented applications. In this chapter we review few of the techniques:

A. Centralized Approach Load Balancing:

In the centralized approach, one node in the system acts as a scheduler and makes all the load balancing decisions. Information is sent from the other nodes to this node. This generates traffic in the grid networks and also increases the scheduling and execution time. Overall this approach increases the cost of a process.

B. Decentralized Approach:

In the decentralized approach, all nodes in the system are involved in the load balancing decisions. It is therefore very costly for each node to obtain and maintain the

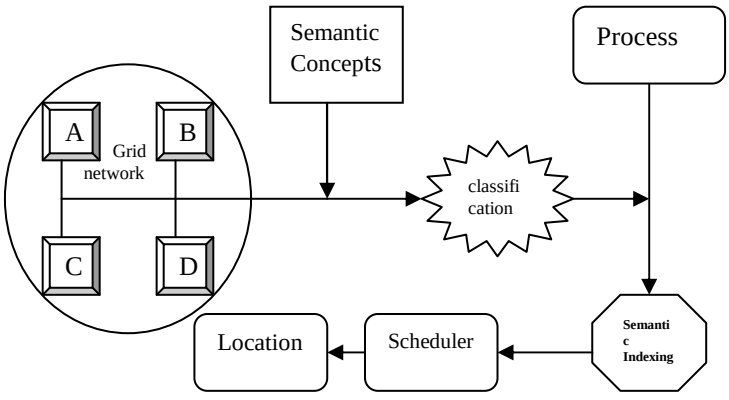

Fig1: Represents overall architecture of the proposed system.

dynamic state information of the whole system. Most decentralized approaches let each node obtains and maintains only partial information locally to make sub-optimal decisions. Static load balancing algorithms assume all information governing load balancing decisions that can include the characteristics of the jobs, the computing nodes and the communication network are known in advance. Load balancing decisions are made deterministically or probabilistically at compile time and remain constant during runtime. The static algorithms have one major disadvantage it assumes that the characteristics of the computing resources and communication network are all known in advance and remain constant. Such an assumption may not apply to a grid environment.

C. *Dynamic Load Balancing:*

Dynamic load balancing algorithms attempt to use the runtime state information

to make more informative load balancing decisions. Undoubtedly, the static approach is easier to implement and has minimal runtime overhead. However, dynamic approaches may result in better performance. One simple load balancing algorithm is the **Best-fit algorithm**. In this algorithm, tasks are assigned according to their order in the queue. Each task is then scheduled to available computing nodes based on the completion time offered by the nodes; the node that completes the task the fastest (taking into account its current load) is chosen.

D. *Histogram Based Scheduling:*

This algorithm is proposed for global load balancing in structured P2P systems. Each node in these systems has two key components: 1) a histogram manager maintains a histogram that reflects a global view of the distribution of the load in the system, and 2) a load-balancing manager that redistributes the load whenever the node becomes overloaded or under loaded. They exploit the routing metadata to partition the P2P network into non overlapping regions corresponding to the histogram buckets.

E. *SCP Based Scheduling:*

In this the application is scheduled according to the availability of data set to execute the process. It maintains the meta data of grid systems and using which it checks for the availability and requirement of data sets to execute the process successfully. After that a grid which contains maximum data will be chosen to execute the process.

III. SCHEDULING:

This step performs the scheduling of the process to help the process to be executed with in short time. The scheduler accepts the input job and retrieves the location of the data from the indexed data. Because of we use semantic indexing , its very easier for the scheduler to retrieve the location of the data objects. This reduce the scheduling time of the processes and also execution time will be less.

A. *Algorithm for Scheduling:*

Step1: Read Input Job

Step2: Identify set of data objects necessary to execute the job.

Step3: compute similarity measure of data objects with semantic concepts.

Step4: Identify the semantic concept with respect to similarity measure.

Step5: retrieve the location of datasets from the indexed results.

Step6: return the results.

At the end of the scheduling process the application will be returned with the location of the grid where the application has to be executed. The query processor will post the process to the returned location and will wait for the result and return the result to the application.

IV. RESULTS AND DISCUSSION:

The final results shows that our proposed scheduling algorithm reduces the overall execution time of the application by reducing the scheduling time and execution time. Our indexing scheme reduces the scheduling time.

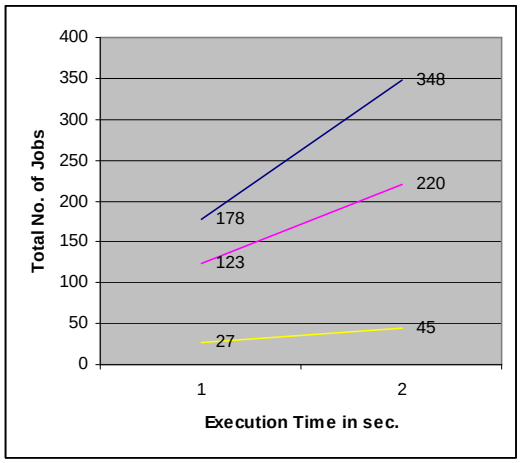

Fig2: shows the analysis of different no. of process and time taken with different algorithm.

Blue line: Histogram based load balancing algorithm

Pink: Scp based scheduling

Yellow: Our algorithm.

V. CONCLUSION:

The proposed method achieves good results and reduces the overall execution time. We further analyze the query execution process to reduce the over all execution time. The proposed SE-AKA is secure against various malicious attacks. The elaborate performance evaluations in terms of communication, computational and storage overhead have been conducted, which demonstrate that the transmission overhead of the whole authentication is considerably reduced, the computational overhead of the HSS and the storage overhead in the serving network can also be decreased, and the bandwidth consumption.

REFERENCES

[1] S3: Scalable, Shareable and Secure P2P Based Data Management System, 2008.
[2] Gnutella, http://www.gnutella.com/, 2008.
[3] BitTorrent,http://www.bittorrent.com/, 2008.
[4] Overnet, http://www.overnet.com, 2008.
[5] V.R. Alvarez, E. Crespo, J.M. Tamarit, Assigning students to course sections using tabu search, Ann. Oper. Res. 96 (2000) 1–16.
[6] G. Barbarosoglu, D. Ozgur, A tabu search algorithm for the vehicle routing problem, Comput. Oper. Res. 26 (1999) 255–270.
[7] R.R. Brooks, S.S. Iyengar, J. Chen, Automatic correlation and calibration of noisy sensor readings using elite genetic algorithms, Artificial Intelligence 84 (1996) 339–354.
[8] I. Forster, C. Kesselman, The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann, 1998.
[9] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Oper. Res. 13 (1986) 533–549.
[10] F. Glover, Tabu search. I, ORSA J. Comput. 1 (1989) 190–206.
[11] F. Glover, Tabu search. II, ORSA J. Computer. 2 (1990) 4–32.
[12] F. Glover, J.P. Kelly, M. Laguna, Genetic algorithms and tabu search: Hybrids for optimization, Comput. Oper. Res. 22 (1995) 111–134.
[13] F. Glover, M. Laguna, Tabu search, in: D. Du, P.M. Pardalos (Eds.), in: Handbook of Combinatorial Optimization, vol. 3, Kluwer Academic Publishers, Dordrecht, 1999, pp. 621–757.