Design of Distributed Multi Agent Servers for Efficient Treatment of Health Care Issues

P.Karthik
Research Scholar
Pondicherry University
Puducherry

S.Bhuvaneswari
Head
Pondicherry University
Karaikal

ABSTRACT
The recent research in the field of health care reveals the fact that, an abysmally low number of people living in rural developing countries have access to specialist care. And particularly country like India[7][8], the factors like poverty, ignorance and vast number of uneducated population further worsen the situation besides lack of awareness of the people about innumerable diseases which are ticking in a rapid phase every day. Due to the above, the rural health suffers the worst as they are often diagnosed with wrong procedures and channelized in wrong direction. Proper earlier diagnosis would have otherwise saved their lives from the effect of dreadful diseases. This paper addresses the above problem by bringing down the gap between people and the specialist care [6] with the support of technology.

General Terms
Peer Agent, Central Agent, Distributed Multi Agent Servers, Unified Information Model and Random Election Algorithm.

Keywords
MA- Multi agent, UIM- Unified Information Model.

1. INTRODUCTION

Hospitals employed expertise doctors; equipped with well infrastructure facilities; being geographically distributed but connected with fairly inexpensive communication networks, is a good choice of application domain for distributed multi agent based deployment solutions. Here, the term Multiagent[1][2][9][11][12][13] refers a collection of loosely coupled, distributed, autonomous and intelligent software system that work cooperatively to perform a task. These software systems are called agents which are capable of performing task beyond their capabilities when working in a coordinated way and in isolation they can do only a partial task with no guarantee of completion of task [2]. Today the modern multispecialty hospitals use local database(s), to store patient related information such as name, address, ward number, type of disease by which he/she is infected, list of medicines consumed by the patient for the disease and tariff details, etc… The disadvantage with this approach is that these data will only be available and accessed with in the corporate hospital network. When the patient or medicos move outside the hospitals then accessing this information is no way possible and hence the patient needs to start the treatment again from the scratch or as the worst case the patient and medicos have to make an extensive travel to the hospital to resume the treatment. Earlier the interconnection and integration of hospitals were thought to be a difficult task for the following reasons.

- *Installation of severs in each hospital and interconnecting them via dedicated communication link were considered to be huge investment of capital cost.*
- *The medicos considered health care treatment as a money making business and they never wanted to share their knowledge openly with public as well as their counter hospitals.*
- *Interoperability between the heterogeneous servers was a challenging task due to the absence of regulations/standards. The proprietary design paradigm followed by the distributed servers further complicated the issue.*

The recent advancements in the fields of Electronics and Computer industry drastically reduced the cost of servers and the communication devices. On the other side; they have considerably increased their processing power and data transfer rate. The attitude of the medicos has been changed towards the public interest which has been evidenced in the recent days such as ehealthforum[1]. The continuous research evolved in the above mentioned industries made interoperability possible through well defined standards and protocols. In a developing country like India the deployment of unique identity AADHAAR[2] to the citizens makes the identification and standardization process easy. The basic idea of the design is provide quality and cost effective health care solutions to the common man with the noble intention to save several millions of innocent people. Expert systems had been initially devised to addresses the above problem but not met huge success because of the nature of isolated database servers.

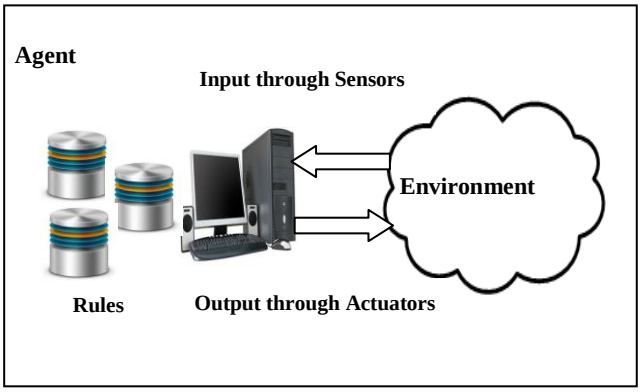

Fig: 1 An Intelligent Agent

1. http://www. ehealthforum.com/
2. http://www. uidai.gov.in

The rest of the paper is organized as follows. In Part 2, we have discussed about related works. In Part 3, we have explained the proposed approach. Part 4 deals with security issues /challenges in the implementation of distributed multi agent servers. In part 5, we have mentioned conclusions and

our future work. In part 6, we have listed out the references we have made.

2. RELATED WORK

In this section we have reviewed some of the peer works done by other authors in the area of distributed computing/database, MA based solutions for other applications, security issues associated with distributed data sharing and Health care solutions using distributed database design.

Van Steen et al [3] have discussed the several of issues associated with distributed computing and data storage. They have mentioned various approaches (RPC, DCE, DCOM, RMI, CORBA and MOM) that have been developed in the area of distributed computing [2][3].

V Vishwanathan et al [2][9][11], proposed solution for integration of data in the power sector using multi agents. As the systems are loosely connected the agents will communicate with other agents only on demand. This way they have drastically reduced the data communication cost. However this approach proposed solution for only static agents.

In paper [4] Ming Li et al proposed solution for exchanging the public health record (PHR) securely in a cloud storage (variant of Distributed Computing) enabling group key sharing[10] with the help of Attribute Based Encryption (ABE). By this approach the user whoever have the decrypt key can able to access the data.

Oana Frunza et al [5] have used Natural Language Processing (NLP) and Machine Learning (ML) techniques, to extract the medical information from short text present in Google health[3], Microsoft Health Vault[4] and Medline[5] health repositories. They have identified semantic relations between two medical entities: diseases and treatments using the above tools and have shown significant performance improvement in the results.

3. THE PROPOSED APPROACH

The proposed approach uses multiple intelligent agents which are loosely interconnected with each other through cost effective communication medium. Each agent is equipped with local database servers that are used to store the particulars of the patients. These agents are usually installed in multi specialty hospitals. The agents are autonomous in nature in the sense that it may have heterogeneous databases for storing the patient information. Among the agents one agent will be acting as a central coordinator which will perform important functions such as user registration, agent registration, user authentication, agent authentication and agent coordination. The central coordinator is provided with additional database servers which will acts as mirror for the primary data server. This is to achieve reliability in the event of disk failures, high data availability and increased concurrency control over stored data.

The proposed design revolves around three major entities namely User, Peer Agent and Central Agent. The following major modules have been used in the effective design of the proposed approach.

3.1 User Registration

Users are the patients who require cost effective and quality health care services. It is presumed that every user is provided with unique ID (In our Design we have used AADHAAR) that will be verified by the Peer Agent/Central Agent upon their registration. Each User will be assigned a password of their own choice. Authentication of the patient is done using the username and password along with the token dynamically supplied by the Central agent. Users are only allowed to view the content and they cannot modify the contents of the central agent.

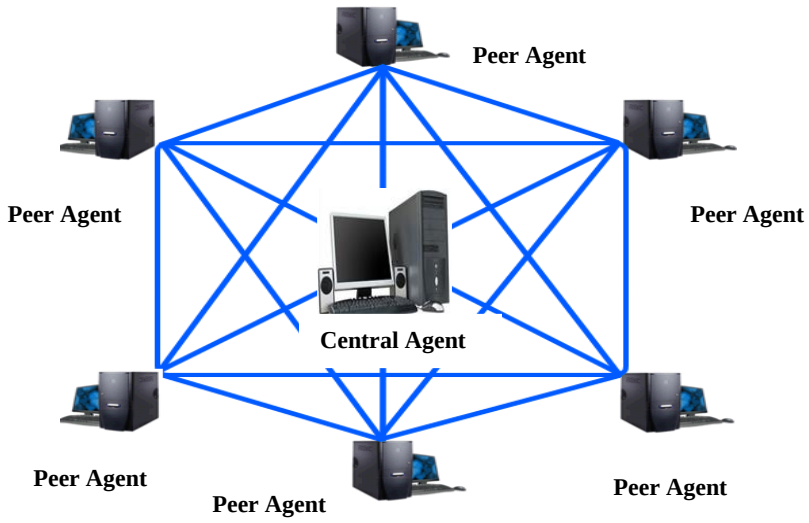

Fig 2: Architecture of Distributed MultiAgent Servers

3. https://www.google.com/health

4. http://www.healthvault.com

5. http://medline.cos.com

The following procedure is used to perform the user registration at the Peer Agent.

Procedure Client_Registration (si , ki , Ti)

Begin

Step1:

S *be the set of users and* ***K*** *be the set of IDs.*

$S=\{x \in T: p(x)\}$

K={y ∈ R: g(y)}

The Algorithm assumes that,

∃ f: p(x)=g(y) ; *where* ***f*** *is an injective function.*

For every ***si ∈ S****; a user defined password is assigned such a way that it satisfies* ***C(x)***

Where, ***C(x)*** *is set of conditions defined by the central agent.*

Step 3:

Authenticate the Peer Agent by means of tags generated through public key encryption along with the help of dynamically selected Auxiliary Peer Agent. This is to prevent any impersonating servers log on into the central server.

Step 4:

On successful completion of the above steps allow Peer Agent to enter the client data into the central agent database.

End;

The successful registration enables the client to retrieve the medical records dynamically from the central agent. In this scenario it is presumed that all the agents were assigned static IP address they know the public keys of other agents. The client can also register himself/herself directly with the central agent. In such circumstances the role of Auxiliary Peer Agent is avoided.

3.2 Peer Agent Authentication

Peer Agents play vital role in the design of distributed multi agent servers. They are mainly used for user registration and also act as local health repository of the clients who are undergoing treatment in the hospital. When the patient is admitted to the hospital it will search for information in its local database using the client ID. In addition to the above, it will also perform search in the central agent to obtain the past history of health records pertaining to the patient admitted. In short we could say the main job of the Peer Agent is to collect up to date information from other agents and populate them it its local database.

The registration of the Peer Agent has to be directly done with central agent. Static IP address and public key are assumed as a prerequisite for the registration. After Successful registration the above entries will be made in the central agent's database. Subsequent to the above, each Peer Agent will be assigned of unique password for authentication.

Authentication of Peer Agent is done at two levels. The following procedures are used for Peer Agent authentication.

Procedure L1_Authentication (IP, pass, token)

Begin

Step 1:

Receive parameters from the Peer Agent such as ***IP***, ***password*** *and* ***token****.*

Step 2:

Perform search in the Central Agent Database for the received parameters. If the entries are found then,

Calculate ***T=f(IP)***

Where ***f ()*** *is a non linear function which will produce a random number using the* ***IP****.*

Step3:

Store the value of ***T*** *in a temporary buffer.*

Step 4:

Split the random number ***T*** *into two unequal pars say* ***T1*** *and* ***T2.***

Calculate ***Tag1 =PEA (T1, Pkey)***

Tag2= PEA (T2, Pkey)

Where, ***PEA*** *stands for Public Encryption Algorithm and* ***Pkey*** *stands for public key.*

Step 5:

Choose the Auxiliary Peer Agent by Random Election Algorithm and compose two messages as;

Msg1= {Encrypted IP of Peer Agent, Tag1}

Msg2= {Encrypted IP of Auxiliary Peer Agent, Tag2}.

Here encryption of ***IP*** *is performed using the public keys of respective machines.*

Step 6:

Send ***Msg1*** *to Auxiliary Peer Agent and* ***Msg2*** *to Peer Agent. Wait for response.*

End;

At the end of L1_authentication (), both Peer Agent and Auxiliary Peer Agent will come to know each other and in turn both agents will wait for mutual response. On receiving Msg1 the Auxiliary Peer Agent will simply forward Msg1 to Peer Agent.

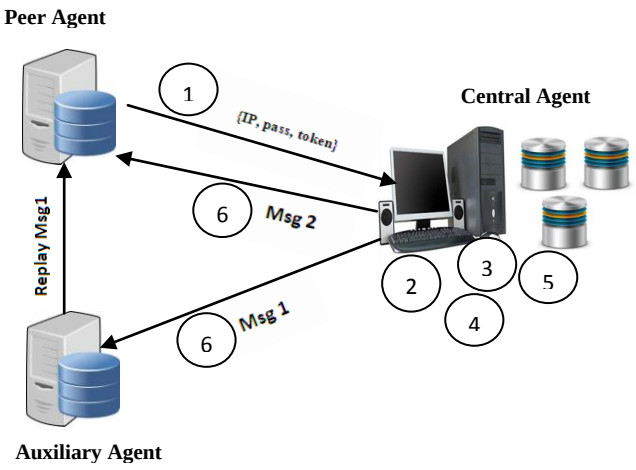

Fig 3: Level 1 Authentication Sequence Diagram

Level 2 authentication mainly concerns with Peer Agent response. The following procedure will be used for generating Peer Agent response.

Procedure L2_Authentication ()

Begin

Step1:

*Receive **Tag1** and **Tag2** through Auxiliary Peer Agent and Central*

T1=PDA (Tag1, PrKey)

T2=PDA (Tag2, PrKey)

*Calculate **RT** as **RT=T1+T2***

Step 2:

*Split the number **RT** into two pars say **T1** and **T2.***

*Calculate **Tag1=PEA (T1, Pkey)***

Tag2= PEA (T2, Pkey)

*Where, **PEA** stands for Public Encryption Algorithm and **Pkey** stands for public key of Central Agent.*

Step 3:

Compose messages as

Msg1= {IP Central Agent, Tag1}

Msg2= {IP of Auxiliary Peer Agent, Tag2}.

Step 4:

*Send **Msg1** to Auxiliary Peer Agent and **Msg2** to Central Agent. Wait for grant permission.*

End;

At the end of level 2 authentication, Central Agent receives encrypted spilt values of ***RT*** which will be decrypted using the private key of Central Agent. The sum of decrypted values is then compared with ***T***. The entire sequence is depicted in the below mentioned diagram.

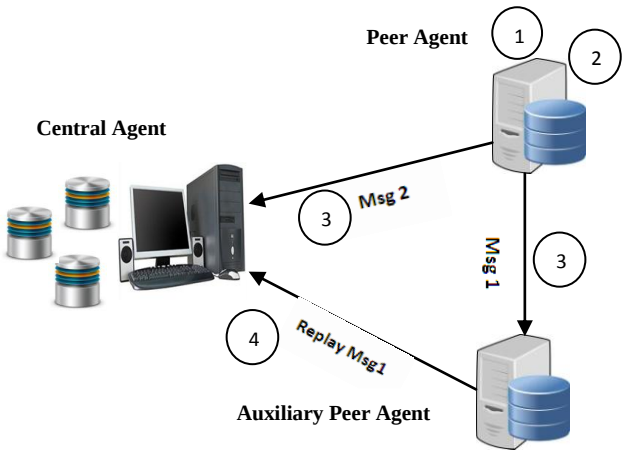

Fig 4: Level 2 Authentication Sequence Diagram

If ***T=RT*** then authentication will be successful. Now the central agent allows the Peer Agent to query the database to get the patient's health records.

3.3 Random Election Algorithm

Choosing the Auxiliary agent in the process Peer Agent authentication, election algorithms plays crucial role. This algorithm first identifies the network in which the Peer Agent is located. Then it will perform search operation in the central agent database to identify the list of other Peer Agents located in the same network domain. This is mainly done to ensure for better security and performance considerations. The following procedure is used in the random election algorithm.

Procedure Random_Election(IP)

Begin

Step 1:

*Parse the **IP** address to identify the first byte.*

Step 2:

*Check first byte to identify the type of network using table listed in **table-1**.*

Step 3:

Perform search in the Central Agent database to identify Other Peer Agents in the same network domain.

Step 4:

*If there are **'n'** Peer Agents in the domain then generate a random number in the range 1 to n.*

Step 5:

Choose the agent as the Auxiliary Peer Agent for authentication.

End;

Class Type	Range of First Octet Value	Higher Order Bit
Class A	1-126	0
Class B	128-191	10
Class C	192-223	110
Class D	224-239	1110
Class E	240-254	1111

Table 1: IP Address Classification

3.4 Unified Information Model

The deployment of the entire design is based on distributed database. As the agents are purely autonomous, each agent may follow different architectural design. Therefore this introduces a new problem called interoperability. This makes the communication between the agents is cumbersome and sometimes impossible. To overcome this problem all the agents are restricted to use unified information model which allows the agents to use similar data models with symmetric metadata. Besides, the proposed design paradigm allows the agent to choose any database software of their own choice.

4. SECURITY ISSUES / CHALLENGES

Security is the major concern in this approach as the application of this design is specific towards health care issues. If any pitfall in the design of security will cause catastrophic effects such as improper/wrong treatment to the patient and as the worst case sometimes it will lead to loss of precious life. Since all the data are stored in the central server, the possibility of attack to the central server is high. To protect the confidential health records of the central server public encryption technique is used. Encryption is done using the public key of respective agent. So that, only the agent can able to know the intelligence of the data using its private key.

4.1 Attacks on Encrypted Communication

Though the communication between the agents is encrypted the intelligence of the data is not always secure. Let us assume the Peer Agent and central agent wants communicate with each other. Both agents need to know the public key of its counterpart. If the request for the public key captured in the middle by an attacker then he can effectively capture the central agent data.

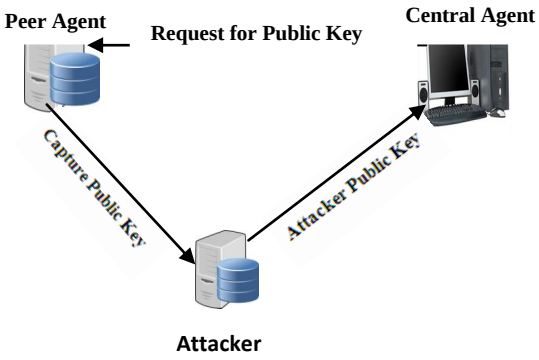

Fig 5: Attack on Encrypted Communication

Here, the response of the Peer Agent i.e Public Key is captured by an attacker by placing himself in the middle. The Attacker now replaces the original public key with attacker public key and sends it to the central agent. The entire message can now be well received by the attacker with his private key. The same message can now be encrypted using hacked public key for sending the same to the Peer Agent. Both sender and receiver do not aware of the data leakage in the middle.

This is because authentication is done with only two entities namely sender and receiver. But the proposed approach uses two level authentications with the help of Auxiliary Peer Agent which ensures that no man is present in the middle. Therefore the data is always safe. The proposed design also addresses problems associated with packet sniffing and spoofing.

5. CONCLUSION AND FUTURE WORK

The proposed design bridges the gap between common people and specialist care. The distributed MultiAgent servers enable both client and medicos to have complete medical health records irrespective of their mobility. This will surely enable the medicos to take better medical decisions before the commencement of actual treatment. From the client's perspective the proposed design enables the patient to have quality treatment irrespective of their mobility. Unnecessary Repetition of diagnosis and Laboratory tests can be avoided which will drastically reduce the cost of health care treatment and thereby allowing the patient to freely move around the globe without worrying about health care issues. Our future work focuses on deployment of the design in the cloud environment. Security is the major threat in the case of cloud storage as the storage control is not in the hands of the user. Remote Data Integrity, Data inconsistency and public verifiability are some core issues which we would like to explore in our future work.

6. REFERENCES

[1] Durfee, E. H. & Montogomery, T. A., "MICE: A Flexible Testbed for Intelligent Coordination Experiments," Proceedings of the 1989 Distributed Artificial Intelligence Workshop, pp. 25-40, 1989.

[2] Z Zhang, JD McCalley, V Vishwanathan , Vasant Honavar., "Multiagent System Solutions for Distributed Computing, Communications, and Data Integration Needs in the Power Industry", Meeting, 2004. IEEE, 2004 – ieeexplore.ieee.org

[3] Van Steen, M. and Tanenbaum, A., Distributed Systems: Principles and Paradigms. Englewood Cliffs, NJ: Prentice Hall (2002).

[4] Ming Li, Shucheng Yu,Yao Zheng, Kui Ren, and Wenjing Lou., " Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption ", ieeexplore.ieee.org /iel5/71/4359390/ 06171175.pdf (2013)

[5] Oana Frunza, Diana Inkpen, and Thomas Tran ., "A Machine Learning Approach for Identifying Disease-Treatment Relations in Short Texts", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 6, JUNE 2011

[6] Brig Pawan Kapoor, VSM, Professor and Head, Department of Hospital Administration, AFMC, Pune – 40 , in his editorial paper (doi: 10.1016/S0377-1237(11)60040-3)

[7] National Vital Statistics Reports, Volume 60, number 4, submitted by the group of people Sherry L. Murphy, B.S.; Jiaquan Xu, M.D., and Kenneth D. Kochanek, M.A., on Deaths: Preliminary Data for 2010 dated January 11, 2012.

[8] National Family Health Survey Reports submitted by Arvind Pandey, Minja Kim Choe,Norman Y. Luther, Damodar Sahu,and Jagdish Chand, on Infant and Child Mortality in India dated Number 11 , December 1998.

[9] V. Vishwanathan, J. McCalley, and V. Honavar, "A Multiagent Infrastructure and Negotiation Framework for Electric Power Systems," Power Tech Proc., 2001 IEEE Porto, Volume: 1, 2001Page(s): 438–443.

[10] K.U.V.Padma, J.Anitha, K.Balaji., "A Novel Multi owner Data sharing Group key protocol",International Journal of Research in Computer and Communication Technology, Vol 2, Issue 10, October- 2013

[11] Z. Zhang, V. Vishwanathan, J. McCalley, and V. Honavar, "A Multiagent Security Economy Decision Support Infrastructure for Deregulated Electric Power Systems", Proc. of 7th Probabilistic Methods Applied to Power Systems (PMAPS), Vol. 1, Page(s): 39-44, Naples, Italy, Sep. 2002.

[12] J. Lind, "Iterative software engineering for multiagent systems – The MASSIVE Method," in "Lecture notes in Artificial Intelligence," Springer-Verlag, Berlin, 2001.

[13] F. M. T. Brazier, B. M. Dunin-Keplicz, N. R. Jennings, J. Treur, "DESIRE: Modelling Multi-Agent Systems In a Compositional Formal Framework", International Journal of Cooperative Information Systems, 6(1), pp 69-94, 1997.