Proceeding of the International Conference on Advanced Science, Engineering and Information Technology 2011

Hotel Equatorial Bangi-Putrajaya, Malaysia, 14 - 15 January 2011

ISBN 978-983-42366-4-9

Security Loop Agents for the Enterprise Applications based on Resource Description Framework

Ahmed Isam Khaleel, Ibraheem T. Nather, Adib M.Monzer Habbal
InterNetWorks Research Group, UUM College of Arts and Sciences,
University Utara Malaysia, 06010 UUM, Sintok, MALAYSIA
alhatemit@gmail.com, ibrahem_m_s@yahoo.com, adib@uum.edu.my

***Abstract*—Security loop-holes can cost a fortune to a large enterprise organization providing e-commerce services. Meanwhile, the enterprise applications have been applied widely to simplify and generate better performance in managing the business tasks. Most of these applications (Enterprise Applications) unable to provide a high level of security due to the new daily threats specially when malicious agents entered into agent platforms and destroyed other active agents during the agent performance for client query. Meanwhile, the security issues in these applications left in the system unintentionally but are intruded intentionally. Hence, this study aims to come out with suitable solution for the existing question on how to secure and platform independent environment for the enterprise applications? By designing architecture to provides a secure and platform independent environment for agents' communication.**

Keywords: Agent systems, J2EE, recommendation method, XML, RDF, OWL, enterprise application.

I. INTRODUCTION

The threats to security are increasing with the emergence of new technologies such as software agents. There have been many attacks in past where malicious agents entered into agent platforms and destroyed other active agents. According to Hogg et al. [7], refers to a real world scenario where malicious agent destroyed the other agents on the platform. According to the author, it will be very critical to focus on security when agents will be used for mission critical systems [3]. In that scenario, a security leak could cause a big harm [6]. Software agents will be an important part of semantic web [11]. The agents will get and understand information from different semantic constructs, for instance ontologies, Resource Description Framework (RDF) and (XML).

Therefore it is important to secure agents and other relevant technologies for safe web services. Multi-agent system is an environment where different agents collaborate to perform a specific task [5]. These kinds of systems are developed to be used in environments such as Internet to perform collaborative tasks. However, this interaction leaves agents in a vulnerable state, where malicious agent can enter to the system. For example, a malicious agent can enter in an agent platform and kill an agent that was used to perform sales. After killing that agent, this malicious agent can process the order and send the payment to wrong party.

Rest of the paper is organized as follows. Issues of the study are presented in section II. Section III presents the proposed architecture. Section VI shows example of the proposed Security Loop over J2MEE. The significant and the expected benefits are presented in section V. Conclusion also presented in section VI followed by the references.

II. ISSUES OF THE STUDY

There exist several techniques and frameworks for agent's communication, but none of those provide cross-platform security [1]. For instance, to encrypt data communication between agents. In their technique both source and destination platforms must have a same cryptography algorithm. This approach negatively affects the performance agent's communication. There are number of users around the globe using the World Wide Web and a number of agents are created by these users [1]. Therefore, in order to reduce the bottlenecks an ad-hoc based authentication is required for agent communication.

The Enterprise applications defined as platform-independent for supporting web application which written in different programming languages, building and deploying Web-based enterprise applications online [8] [11]. The J2EE platform consists of a set of services, APIs, and protocols that provide the functionality for developing multitier, Web-based applications.

The main enterprise application features can be addressed into the following:

- Working together with the HTML based application that consists on Java Server Pages and to build the HTML web contents or other formatted data for the client.
- Provide external storage platforms' that are transparent to the author.
- Provide database connectivity, for managing and classifying the database contents.

These technologies are the important constituents of web services. It is therefore very likely that these web services will be agent based in the near future. The success of enterprise application will highly rely on the implementation and usage of these web services. Agents can use intelligent collaborations in order to achieve global optimization while adhering to local requirements.

Figure 1 presents the J2EE communication network among its components.

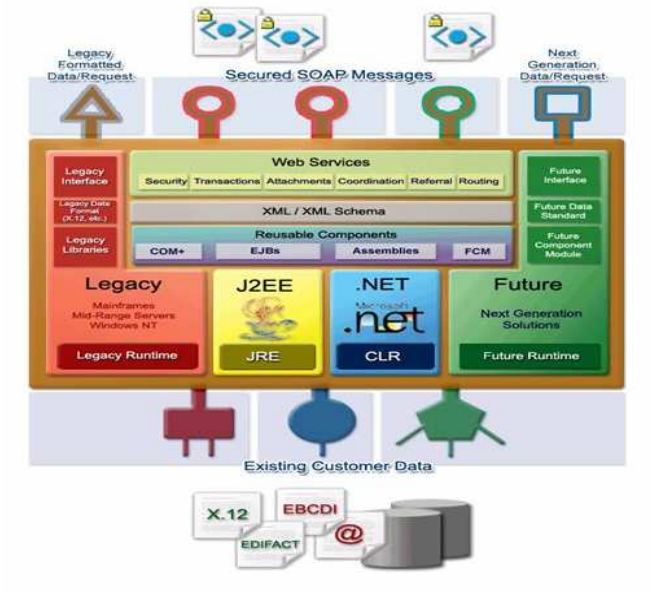

Fig 1. Enterprise application communication network

The current adopting of the new technology have brought a new ideal integration for securing and simplifying the data sharing for all components of enterprise applications [9]. The elements of enterprise application which can be possibly configured within the container as stated in Figure 2 the following:

- HTML, JSP, Servlets and other static binaries which are accessed via a browser-based client.
- Regular non-browser applications, both Java-based and otherwise, which can access enterprise application components like EJBs directly.
- Enterprise services like, messaging/EAI, Data repositories, third-party enterprise application - based products, etc. which needs to accept and verify credentials seamlessly.

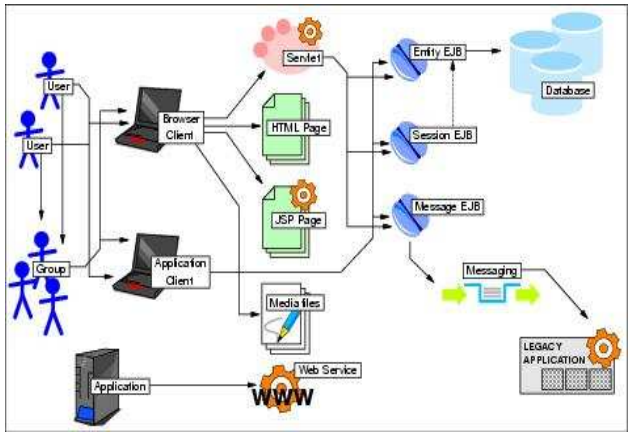

Fig 2. The security process in enterprise application

III. The Proposed Architecture

As known, the representing and accessing of the web contents among platforms are determined to be a more recent innovation; most of this representation involves the use of other techniques such as (RDF, XML, and Ontology web language OWL) these technologies works together to link systems together. Enterprise application platform independent facing several security problems in data sharing and accessing which enable web services to work across low level of security. However, the communication process in these platforms (Enterprise application) from the client to the service uses certain technology that helps to translate the client data and assign its security level based XML as the common language. This allows one application to call the services of another application over the network by sending an XML message to it.

Thus, our proposed architecture will be more efficient in a way that there is no need for encrypted agent's communication, which reduces the processing and communication time. Also our proposed architecture will be platform independent because there is no need to maintain standards for cross-platform agents' communication security.

In a pervasive environment, trust can be used for collaboration among devices. Trust can be computed automatically without user interference on the basis of direct and indirect communication [2]. In the direct communication or observation mode the device user's interaction history is considered. For this purpose a trust value is assigned to each identity in the trust database [12]. There exist some formulas such as (Observations and recommendations) that use to calculate the single trust value for the user on the basis of observations and recommendations [2].

This study will apply the recommendations technique which aims to specify a degree of trust for each person in the network, for automating trust, which is also called indirect communication [4]. Therefore the observation and recommendation are used together to generate a trust value for a user. Given a user trust value, a trust category is assigned to user with a value of low, medium or high. The trust values should be regularly monitored because when a new recommendation is received new trust value is compared with its old value and trust database is updated by the enterprise application services for single and multi accessing which operate the use access accordingly.

Recommendations are another method of automating trust, which is also called indirect communication [16].

Therefore the observation is used together to generate a trust value for a user. Given a user trust value, a trust category is assigned to user with a value of low, medium or high. The access rights distribution is performed on the basis of the category value. The trust values should be regularly monitored because when a new recommendation is received new trust value is compared with its old value and trust database is updated by update trust category accordingly.

Figure3 and 4 presents the type of trust over enterprise applications which model the logical relationship between the nodes. These nods will be classified into several groups such as:

- Process Request Group: A request for a service group composed of nodes, node I and node n.
- Register Level Group Provider Group: to provide a service in the network of nodes that comprises the group, as these nodes share certain files, or the provision of certain goods purchases.
- Trust Level Group: trust nodes that comprise the group, node m1, node m2 and node m3.
- Save trust nodes Group: trust network, trust in other nodes on the path formed by the agent.

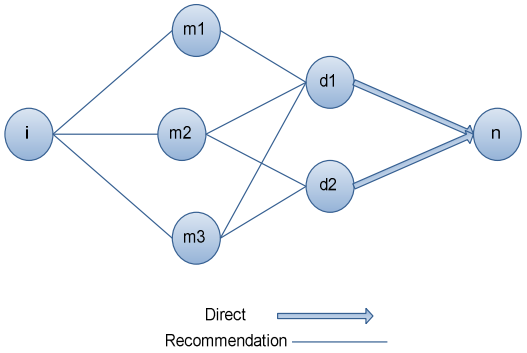

Fig 3. Two type of trust

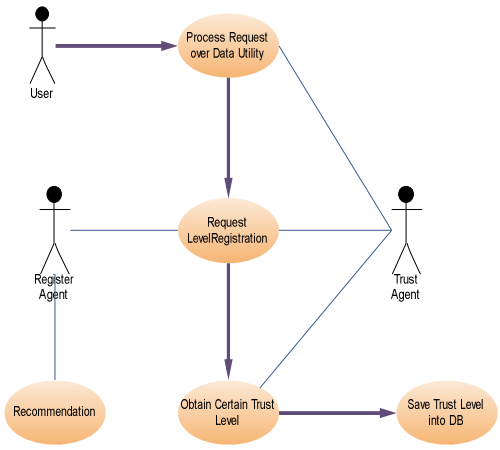

Fig 4. Truest Network based Recommendation and Observation

IV. Example of the proposed Security Loop over J2EE

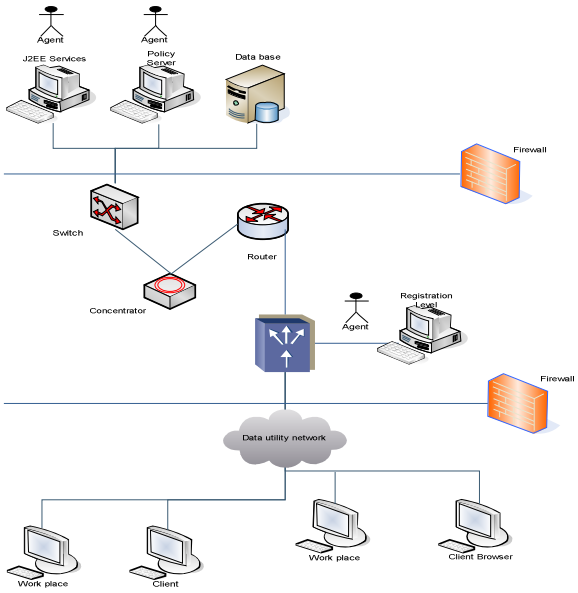

Fig 5. The proposed Security Loop Architecture

In Figure 5 an agent for registration level outside the environment sends a request to server for registration, server registers it with the lowest security level. With the passage of time the agent becomes more trustworthy based on observations and recommendations.

Delegation is the most important feature in our proposed mechanism through which an agent can delegate set of its rights to another agent for specific period of time. In summary of the whole discussion, we proposed a multi-layered security level mechanism whereby an agent enters in the environment with a low level of security and achieves the higher level of security as it survives in the environment.

V. The Significant

The expected benefits from the proposed security architecture can be determined the following:

- **Manage user access by level or authority** this could be done by allowing administrators or trusted clients to access and share the information across platform based on the retrieved recommendation. Furthermore, this feature will helps to assigns different authorities to different administrators based on specific levels that identified by agent.
- **Determine the Client behavior** Moreover, the proposed architecture can be capable of customizing the client behaviors based on the security policy contents that over legal clients to use its services and guard against unauthorized use.
- **Provide a High reliability** Adopting agent systems will helps to simplify the communication performance between client and server.

VI. CONCLUSION

This study aimed to provide a reliable architecture for increasing the security level based on recommendation method. Meanwhile, the best way for representing and organizing the security for all web resources based platform involves the use of a centralized, identity-centric web security system along with a certain language for translating the client request into understandable order based policy enforcement point. Finally, this study was succeeded to determine the working process of enterprise application platform among web services; also expected benefits were reported in term of the adopting of agent technology and recommendation method for assigning the security level for the clients

REFERENCES

[1] V. Bindiganavale, and J. Ouyang,"Role Based Access Control in Enterprise Application Security Administration and User Management, pp.111-117, IEEE, 2006

[2] M. Youna, and S. Nawaz, "Distributed Trust Based Access Control Architecture to Induce Security in Pervasive Computing" Computer Engineering Department, EME College, NUST Pakistan 2009

[3] S. Kagal, T. Finin, and Y. Peng, "A Framework for Distributed Trust Management", in Proceedings of IJCAI-01, Workshop on Autonomy, Delegation and Control, Montreal, Canada, 2001

[4] Foundation for Intelligent Physical Agents. http://www.fipa.org

[5] S. Stojanov, I. Ganchev, I. Popchev, M. O'Droma, and E. Doychev, "An Approach for the Development of Agent- Oriented Distributed eLearning Center," presented at International Conference on Computer Systems and Technologies (CompSysTech), Varna, Bulgaria, 2005

[6] Y. Li, W. Shen, and H. Ghenniwa, "Agent-Based Web Services Framework and Development Environment," *Computational Intelligence*, vol. 20, pp. 678-692, 2004

[7] J. Hogg, D. Smith, F. Chong, D. Taylor, L. Wall, and P. Slater, "Web service security: Scenarios, patterns, and implementation guidance for Web Services Enhancements (WSE) 3.0, "Microsoft Press, March 2006

[8] M. Schumacher, E. Fernandez-Buglioni, D. Hybertson, F. Buschmann, and P. Sommerlad, "Security patterns: Integrating security and systems engineering," John Wiley and Sons, December 2005

[9] Networked Digital Library of Theses and Dissertations Homepage, http://www.ndltd.org (current October 2009)

[10] Open Archives Initiative Tools, http://www.openarchives.org/pmh/tools/tools.php (current October 2009)

[11] F. Almenarez, A. Marin, C. Campo, and C. Garcia, "TrustAC: Trust-Based Access Control for Pervasive Devices", International conference of Security in pervasive computing, Vol. 3450, pp. 225-238, 2005

[12] M. Haque, and S. Iqbal, "Security in Pervasive Computing: Current Status and Open Issues", International Journal of Network Security, Vol. 3, No. 3, pp.203–214, 2006