An Insider Attacks Immune Mobile Agent System using Context and Insider Aware Policy Language

J. J. Adri Jovin
Assistant Professor, Department of Computer Science and Engineering
SriGuru Institute of Technology
Coimbatore, INDIA.

M. Marikkannan
Assistant Professor, Department of Computer Science and Engineering
Institute of Road and Transport Technology
Erode, INDIA.

***Abstract*— Mobile Agent technology has made a revolution in the domain of Distributed Computing. However they are prone to a lot of external threats. In this paper, we address the Insider threats which are caused due to the Malicious Mobile Agents present inside a Mobile Agent Environment. We design an Access Control Framework enabled Mobile Agent platform in order to overcome the Insider threats. This Mobile Agent platform works on the basis of a Trust-Based Environment. We use the Context and Insider Aware Policy Language to make the Mobile Agent Environment suitable to evaluate the Mobile Agents accessing the system. From the theoretical inferences made, it is evident that the Access Control Framework enabled Mobile Agent platform could control Insider attacks to a maximum extent.**

Keywords- Mobile Agent; Insider attack; Context and Insider Aware Policy Language;Trust based systems; CIAPL.

I. INTRODUCTION

Mobile Agent Technology is one of the most promising technologies in Distributed Computing. Mobile Agents are software modules which could be provided privilege by the users to carryout tasks on behalf of them. Their capability to move from one system or environment to another has made them to be used in various domains of computing in which the systems are available in distributed environment. In spite of the benefits possessed by the Mobile Agents, there are certain drawbacks due to security issues. Various security issues have been found out and are addressed. In this paper, we focus on the Insider Attacks which are very hard to be detected. We use the Context and Insider Aware Policy Language [1] to outline such a system.

The paper is organized as follows: Section 2 deals the works related to the insider attacks and Access Control. Section 3 explains the usage of Context and Insider Aware Policy Language to identify Insider Attacks. Section 4 describes the experimentation and section 5 concludes the paper.

II. RELATED WORKS

Access Control mechanisms which are automated play a vital role in resource utilization in large systems. The report produced by Moore et al. has made a number of findings about the factors that contribute to an insider attack [2]. The same problem is applicable in case of Mobile Agents also, since the Mobile Agents hold the privileges of a user and works on behalf of a user. The problem of insiders is mostly associated with the roles given to them by the organization. Hence Role-Based Access Control (RBAC) plays a vital role in the prevention and detection of insider attacks [3].

Brackney and Anderson define an insider to be “someone with access, privilege, or knowledge of information systems and services” and a definition of the insider problemas “malevolent (or possibly inadvertent) actions by an already trusted person with access to sensitive information and information systems” [4]. Distinctly, Patzakis defines an insider as “anyone operating inside the security perimeter” [5].

[6] proposes a model based on Workflow Authorization Model (WAM) and Role Based Access Control to avoid unauthorized access. This method is completely based on the workflow specification for a particular object and the operations it performs. This work provides Flexible Workflow Authorization Templates for providing a user flexible authorization framework, but is completely dependent on the Workflow Specification provided to it. This Flexible Access Control Model for Dynamic Workflow is well suited to restrict unauthorized access which is initiated by external forces. However, it could also be observable that this mechanism could not be used for authorization threats which arise internally in an organization.

The Key Management and Access Control scheme for Mobile Agent proposed in [7] provides a complete Access Control solution in the Distributed Computing domain especially while using Mobile Agents. This mechanism helps the Mobile Agent to be resistant against Reverse Attack, Conspiracy Attack and External Collective Attack. The two schemes discussed in this work effectively manages the keys thereby makes the access control mechanism effective. This mechanism though is provably effective in managing the access control, could not prove itself effective in case of insider attacks. The Attribute-based Access Control System [8] is dynamic and incorporates privacy preferences of the service requestor in the access control process. Also, this system separates the Policy Decision Point from the service provider. This often increases the threat on the PDP being attacked by some malicious agent. The system provides security in case of an external intruder, but cannot manage, if the intruder is an insider.

Access Control Model based on RDB Security Policy [9] is used to secure the ontologies based on OWL. The RDB based security model clearly provides a persistent control over the access of data in the ontology. [10] proposes an Access Control mechanism using Intelligent Agents in Federated Information Systems. The problems such as access over local heterogeneous type of data and local autonomy of the system are overcome using the Intelligent Agents. These agents are governed by the personnel inside an organization. This increases the threat of data being acquired by an insider, without the knowledge of other personnel inside the organization.

Various Access Control techniques like Scalable Access Control, Fine-Grained Access Control and Active Access Control [11] are being employed to prevent Insider Attacks in an Organization. These are access control mechanisms which are employed focusing over the scalability, granularity and context-awareness. Various Schemas [12] are also being used to reduce the complexity of access control mechanisms.

In most cases, it could be observed that the Access Control Mechanisms used are designed to control the external access over the data, especially in a distributed environment. It could also be observed that the Insider Attacks are less focused. In our work, we concentrate mainly over the Insider Attacks. We use a formal method to overcome the threats which arise because of those who are within the organization itself.

III. Proposed Solution

The Insider Attacks are mostly associated with the access control systems inside the Mobile Agent Platform. Hence, the Insider Attacks could be addressed only by means of efficient policy framing for the Access Control inside the system. The major challenge is to develop the system such that it could detect, record and collate all events that would indicate the suspicious activity from authorized users.

A. Terminologies

An assumption, that the components of the Mobile Agent Platform behave in a particular way based on trustworthiness. Those components are collectively known as the 'trust computing base' (TCB). Another assumption that the Authorized Mobile Agents are 'trusted' is also made. This means that the Authorized Mobile Agents do not abuse their privileges. Though the Authorized Mobile Agents are trusted, some of them may not be 'trustworthy'. In terms of formal language, the insider problem arises when the set of trusted users, in some context, is not equal to the set of trustworthy users. In short, it could be expressed as

$$\{trusted\} \neq \{trustworthy\}$$

'Degree of Insiderness' of a Mobile Agent is the threat level posed by the Mobile Agent. The higher the privilege of a Mobile Agent, the higher is the risk it represents. There are two tools used to compute trustworthiness namely trust-management and reputation management systems. A malicious intruder Agent may manipulate the audit records to hide the evidence of breaches of trust to maintain his trustworthiness.

B. Context and Insider Aware Policy Language (CIAPL)

It is evident from the previous section that the insider problem arises because of the divergence between trust and trustworthiness. An Agent authenticates with the Platform by means of Agent ID, Host ID, role identifier etc. A security context will be associated with all the resources accessed by the Mobile Agent and this context would be used to evaluate whether subsequent actions are authorized or not. The security context may be considered as a function comprising of both the Agent ID and the method of authentication. It may even include more number of security parameters.

When the security context is used to determine the actions with which the associated user is authorized makes trust more applicable. The security context may contain attributes like roles which are directly associated with the authorization of an Agent towards a resource. For example, a 'Manager' Agent may be authorized to perform certain actions based on the Authorization Policy. The CIAPL is a general-purpose language for specifying access control policies which could consider factors such as risk and trustworthiness of an authorized Agent into account. The various features of CIAPL with respect to an Mobile Agent Environment is discussed below.

1) Context and Request Predicates

The access request is usually considered as a tuple of form (s, o, a, c) where

s denotes the subject. (e.g) user
o denotes the object. (e.g) protected resources
a denotes the action. (e.g) read, write
c denotes the contextual information (e.g) whether the requestor is accessing the resource remotely.
Sets of tuples are considered as subsets X of space $S \times O \times A \times C$ of access requests. The subset X is called as request predicate. The request predicates are dependent on the context C which is named context predicate.

Provided $X \subseteq S \times O \times A \times C$, we use p_X^+ to denote the policy that authorizes all requests in X and denies all other. Similarly, p_X^- is used to denote a policy which authorizes a request if and only if it is not in X.

The requirements for policy-based access control that can address insider threats can be described as follows:

i) The ability to support rich types of policy decisions beyond "deny" and "grant", which takes into accounts the inconsistencies and errors.
ii) The ability to declare Context and Request predicates.

The ability to transform access control policy p into a set of access requests X .

2) Policy Transformation

The system must be able to offer policy transformation using some declarative programming technique. The first and foremost requirement specified in the previous section allows considering a wide variety of functions and inputs as policies. Therefore a policy might be composed of sub-policies which may return quantitative decisions.

The next requirement suggests the use of request predicates and propositional-logic connectives, so that expressions can be formed as shown below.

$$Manager \wedge Onduty \wedge \neg Weekend$$

In the above expression $Weekend$ is a context predicate. It the subject, here the role $Manager$, requests a resource, the main emphasis is over the role rather than the type of resource. The policy could be formulated as follows

$$Grant \quad if \quad Manager \wedge Onduty \wedge \neg Weekend$$

This could be declared p_X^+ as discussed above. The above policy allows if all the constraints in the request predicate are satisfied.

The third requirement allows identifying sets of access requests that arise from the policies. Consider two primitives

$$p @ Denies \qquad\qquad p @ Grants$$

Given an access control policy p , declares request predicates that capture the set of policies denied or granted by p respectively.

An Agent possessing the role $Manager$ can work at weekend but are not supposed to assign or increment the salary. If any such activity is found the agent could be considered an Insider and hence, the policy would be

$$insider = (manager \wedge assign \wedge payIncrease) \vee$$
$$(\neg manager \wedge rate \wedge managerPerformance)$$

In the above case, the manager must be an outsider in order to evaluate the performance. Also, the trustworthiness of an Agent needs to be considered while analyzing it. The policy language may be stated as follows.

```
  bool tooRisky(R: req; riskThreshold : double){
  return (cost(R)/ trustworthiness(R.subject) >
riskThreshold)
  }
```

The above module takes a single request R and Risk Threshold, which is the maximum tolerable risk that the Access control system, can face, as its input. This return a Boolean which either grants access to the Mobile Agent or denies it.

IV. Experimentation and Results

The experimentation is done using a customized simulator which is designed to check the formal correctness of our method and to simulate the insider attack and check it with some other existing Access Control mechanism. We compare our model with the existing Flexible Access Control Model for Dynamic Workflow (FACMDW). Initially, we setup the simulator to verify the formal correctness of the proposed model using the theorem-verifier module inside it. The various formal arguments and the assignment made over them are fed to the simulator. It is observed from the results obtained from the theorem-verifier that the method is formally correct.

Now, the simulator simulates various attacks over the access control mechanism of the system to test the resistibility of the mechanism. The simulator is programmed especially to simulate insider attacks and find the attack detection rate of the proposed mechanism. Initially we simulate a network comprising of 30 systems. Each system is capable of supporting Mobile Agents. The FACMDW model is implied into the system and tested for immunity against insider attacks. Later the proposed method is implied into the system. The test results based on the immunity rate is displayed in Figure 1.

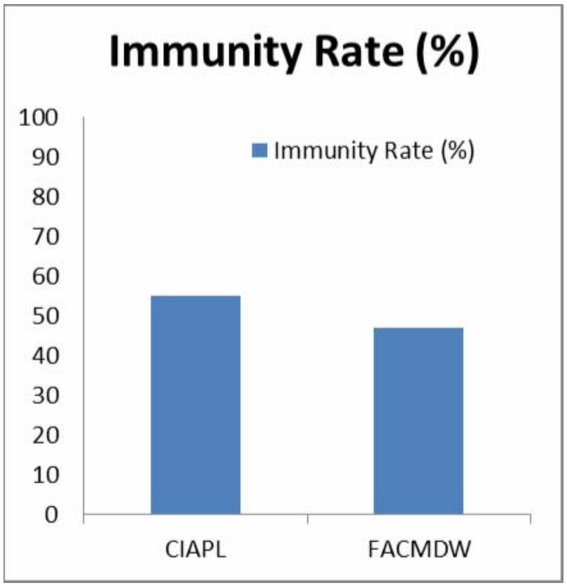


Figure 1. CIAPL vs FACMDW(Immunity Rate)

From the figure, it could be seen that the Immunity Rate of our proposed system is more than that of the system using FACMDW. This immunity rate is with respect to that of insider attacks and does not focus on any other type of unauthorized access. Therefore, it is evident that the proposed system is more effective in case of Insider Attacks rather than that of FACMDW.

V. Conclusion and Future Work

In this paper, we use the Context and Insider Aware Policy Language to achieve immunity against Insider Attacks. Moreover an Access Control Framework is established inside the Mobile Agent Platform to make it possible for the CIAPL to act over it.

In future, the formal methods that are used in this work shall be implemented in real-time systems for verification of the results in a real-time environment.

References

[1] Jason Crampton and Michael Huth, "Towards an Access-Control Framework for Countering Insider Threats", Insider Threats in Cyber Security and Beyond. Springer, Heidelberg , 2010.

[2] Moore, A. P., D. M. Cappelli, and R. F. Trzeciak, "The "Big Picture" of Insider IT Sabotage Across U.S. Critical Infrastructures", Technical Report CMU/SEI-2008-TR-009, ESCTR-2008-009, Carnegie Mellon University, May 2008.

[3] Sandhu, R. S., E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-Based Access Control Models", IEEE Computer 29(2): pp. 38–47, 1996.

[4] Brackney, R., and R. Anderson, "Understanding the Insider Threat", Proc. of a March 2004 Workshop, RAND Corp., Santa Monica, California, March 2004.

[5] Patzakis, J., " New Incident Response Best Practice: Patch and Proceed is No Longer Acceptable Incident Response Procedure" , Guidance Software, Pasadena, California, September 2003.

[6] Le Yang and Yongsun Choi, " A Flexible Access Control Model for Dynamic Workflow using Extended WAM and RBAC" , CSCWD 2007, LNCS 5236, pp. 488-497, 2008.

[7] Pan J.Y., Chen T.L. and Chen T. S., "A Novel Key Management and Access Control Scheme for Mobile Agent" , ICIC 2006, LNAI 4114, pp. 334-345, 2006.

[8] Jan Kotler, Rolf Schillinger and Gunther Pernul, " A Privacy-Enhanced Attribute Based Access Control System" , Data and Applications Security, LNCS 4602, pp. 129-143, 2007.

[9] Jeong, Jing and Baik, " Access Control Model Based on RDB Security Policy for OWL Ontology" , ICCS 2007, Part II, LNCS 4488, pp. 720-727, 2007.

[10] Maranda, " Access Control of Federated Information Systems", EuroISI 2008, LNCS 5376, pp. 119-130, 2008.

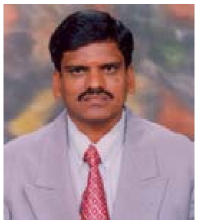
[11] Joon S. Park and Joseph Giordano, " Access Control Requirements for Preventing Insider Threats" , ISI 2006, LNCS 3975, pp. 529-534, 2006.

[12] Ray and Muller, " Using Schemas to simplify Access Control for XML Documents" , ICDCIT 2004, LNCS 3347, pp. 363-368, 2004.

AUTHORS PROFILE

J.J. ADRI JOVIN received B.Tech degree in Information Technology from Anna University, Chennai in 2009. He received his M.Tech degree in Information Technology from Anna University of Technology in 2011. Currently he is pursuing his PhD in Anna University of Technology, Coimbatore. He is working as an Assistant Professor in SriGuru Institute of Technology, Coimbatore, INDIA.

His research interests include Agent Based Intelligent Systems, Network Security and Object Oriented Systems Design. He is a life member in ISTE and ACCS.



M. MARIKKANNAN received the B.E degree in Computer Science and Engineering from Government College of Engineering, Tirunelveli, India in 1994. He received M.E. degree and Ph.D in computer science and engineering from College of Engineering, Guindy, Anna University, Chennai, India in 1999 and 2009 respectively. Currently, he is an Assistant Professor in the Department of Computer Science and Engineering, Institute of Road and Transport Technology (IRTT), Erode, India.

His area of research interests include temporal database management Systems, Object oriented systems, Data Mining, Wireless networks with security.