The Nucleus **48**, No. 2 (2011) 121-127

The Nucleus
A Quarterly Scientific Journal of Pakistan
Atomic Energy Commission

NCLEAM, ISSN 0029-5698

A MULTIAGENT SYSTEM BASED AUTOMATED PROFILING MODEL TO AVOID INSIDER THREAT

*G. ALI, N.A. SHAIKH, A.W. SHAIKH and Z.A. SHAIKH[1]

Department of Computer Science, Shah Abdul Latif University, Khairpur, Pakistan

[1]National University of Computer and Emerging Sciences, Karachi, Pakistan

(Received May 24, 2010 and accepted in revised form March 14, 2011)

The proposed model builds and maintains the profile of all insiders to detect and avoid threat. Software agents are autonomously working to record all activities of the authenticated users. The profile is compared with the policy of the organization and insider’s profile is marked acceptable or suspicious. This will lead to a proper mechanism to protect organizations assets against threat. The model is generic, adaptable and works in most of the organizations. It follows the renowned agent standard of Foundation for Intelligent Physical Agents (FIPA) as agents built on other platforms can interact with the agents developed on the proposed model. The profiling models exist but agents based autonomous models have never been implemented and tested before. The testing and execution environment for Multiagent systems has been developed by ourselves and its compatibility with other agent models have also been checked.

Keywords: Insider threat, Software agents, Multiagent systems, Profiling, Behaviour monitoring.

1. Introduction

The problem of Insider Threat has been addressed in this paper and the Profiling is the proposed solution for that. The fundamental definition of profiling is to monitor the activities of an employee within an organization. During investigation and implementation of the vulnerability assessment model it is learnt that insider threat is in fact a giant dilemma where both technology and user behavior must be addressed [1]. Yet in the existence of sophisticated expertise network remains insecure because of capricious human behavior that is always threat to the organization. Therefore, in a corporate environment user’s behavior must be monitored to know the actual personality. There is always need to know what user really is doing in the organization, whether user is going beyond limits or working within the policy of the organization [2]. There is need to observe whether user is focusing on the prime responsibility or wasting time and money of the organization [3]. A complete profile of a user and an autonomous approach to handle it is needed to achieve the goal. Profiling for security through software agents is the solution that is presented in the paper. At low level, organizations are using different kinds of technology, protocols, procedural security measures, but an agent based autonomous system at high level is needed for profiling to monitor user activities in an organization [4].

Addressing following questions will support to build an efficient model against insider threat. For example the activities that a user is performing in the organization, are in accordance with organization’s policy or not? Whether user’s behavior is normal or suspicious? Whether user is certified to do so or not? Whether user is crossing limitations or remains within them? Whether user comes into view from the particular machine or coming from other machines too? How much someone is destructive for the organization? The Agent Collaborative Environment on NET (ACENET) scores every user of the organization and maintains a detailed profile. It is really hard to determine whether a legitimate user is doing anything malicious activity. Expectantly such activity would stand out as strange when compared to the user's routine behavior. This kind of theory was available but not done experimentally [5]. As

[*] Corresponding author : ghulam.ali@salu.edu.pk

mentioned earlier that profiling models exist but no implementation has been done in this regard. Therefore, related work in context of development and testing is no more available [6].

2. Architectural Design

ACENET is adaptable and can be easily deployed in any corporate environment. At application layer any agent can be designed within a shorter period of time on user demand because the abstraction has been provided; only behavior of the agent is to be defined. Agents have been designed as service on the top layers of ACENET. The available agents create profile of the user and start monitor activities autonomously. The threats have been categorized and for each category agents have been designed to monitor behavior of the users. The architecture of the ACENET is capable to adopt any given policy in accordance to any organization [7].

Agent and the execution environment are main factors of this model where agents operate to solve the problem through predetermined and learning based routines. ACENET is developed package to provide suitable environment over computer networks where various agents can live and perform other activities autonomously. Moreover, these agents can communicate with each other as well. At higher level this communication takes place just by message passing but internally socket based communication is underway [8]. The agent designing is based on the classification of server and client side environment.

The description and activities detail of the agents running over ACENET is given below.

2.1 Server Supporting Agent Designing

These are the agents that reside on the system administrator machine and manage the framework across the network. Currently three kinds of the instances (agents) are developed:

2.1.1. Manager Agent

A powerful and most efficient agent that receives the messages from client agents and forwards to the database in the view of predefined organizational policy. It performs multiple insertions actively and maintains the database datewise.

2.1.2. Alarm Agent Server

Alarm Agent is in fact an idea of designing agents on client’s requirement to generate alerts in case of severe threat. Organization’s policy will decide the sensitivity of the most harmful action of the insider user. For example if organization defines external device attachment as the major violation, then the alarm agent is capable to get an image of that activity from the client agent and propose immediate response to that.

2.1.3. Analyzer Agent

The third key component of server side agent is capable of reading the data from database and transforms them into reportable form for viewing of the administrator. This agent is supportive as it could provide decision support for the organization through its analysis.

2.2 Client Supporting Agent Designing

These are the agents that remain active on client end. They get activated right at the moment when any one logins the machine and starts work and some of them instantiated on trigger predefined to them. Three kinds of the instances are given below:

2.2.1. Profiling Agent

This agent keeps monitoring the non-cyber activities performing by the logged machine and forwards towards the manager agent. The monitored activities contain running processes, applications, system login, logout time, removable media usage, printer usage etc. This agent actually governs a set of agents who are dedicated to perform specific tasks indicated with their names. These agents are: Active application monitoring agent, Process monitoring agent, Session monitoring agent, Print activities monitoring agent, and Removable storage monitoring agent, etc.

2.2.2. Cyber Agent

This agent keeps the track of all visited websites from the machine and sends them towards the manager agent in order to put them into the database.

2.2.3. Alarm Agent Client

This dynamic agent is activated when user violates some sensitive rule of the organization. Therefore, it is named as online monitoring. This agent sends screenshot at the instance when user performs most threatening activity. Backend database is used to keep the record in manageable form as it could be analyzed and even predict in future.

There are four main components of the model such as client, server, agent and database. Agent (system profile agent in the model) resides on the client that builds profile of the user and monitors activities. The Manager agent, residing on the server, collects information from the system profile agents, residing at various clients, on the entire network and sends these reports to database. Database stores updated profiles of the users sent by the manager agent. Administrator retrieves the profiles of the users from the database, creates reports and analyze manually in the light of the policy of the organization. The model has been developed to ascertain that the user is authorized to access the confidential data and sensitive assets within organization, are authenticated through their identity and password. It is assumed that all users as trusted because they all are authenticated and authorized by the system to access the resources.

As user logs in the system, the system profile agent is activated and starts monitoring the behavior. The system profile agent monitors the activities of the user and sends time to time updates to the manager agent at server. As server agent receives the data from the system profile agent, it simply inserts that data in the profile of the concerned user in the database. The system profile agent is proactive, whenever an insider intends to carry out a malicious act, the system profile agent will alarm to manager as well administrator as the threat may be avoided before occur. For example an organization does not authorize to plug external devices to a specified level of users. If a user plugs external device to the system, the system profile agent will alert the manager agent as well as administrator to avoid the threat. After gathering profile of the users, necessary reports are generated as the suspicious activities may be detected. To response against critical threat, if user indulges in suspicious activity then immediately block of the system will take place. User's account will be disabled and system or the user will be isolated. To avoid user's interruption to the agents some protection measures will be proposed in future.

2.3 Building Profile Through Activities Monitoring

To generate and maintain profile of the activities of all users browsing is assumed as the main source. Browsing has been further divided into windows resources browsing and internet resources browsing [9].

2.3.1. Windows Resources Browsing

What and how the resources of the systems and network have been visited. This browsing includes following activities to be monitored.

- Login/Logout time: Total session time will be recoded
- Processes/applications: The running time and the names of the applications and the processes are tracked in this activity
- The shared resources access (files, printers, folders): Remote shared resources such as accessing files, folders, etc. will be tracked
- Time to time Screenshots of the user's machine will be taken as evidence
- Printer usage: The printing detail of the documents will be tracked
- External device in/out: The usage of external devices such as USB for copy in or out will be tracked

2.3.2. Internet Browsing

Internet browsing activities of the users will be monitored in this browsing. This browsing includes following activities to be monitored, shown in Figure 1.

- Names of websites/IP addresses user visited
- Website visiting Frequency and duration
- User's most interesting website names
- Category of websites, user visited
- Web-mail access and uploaded files as attachment

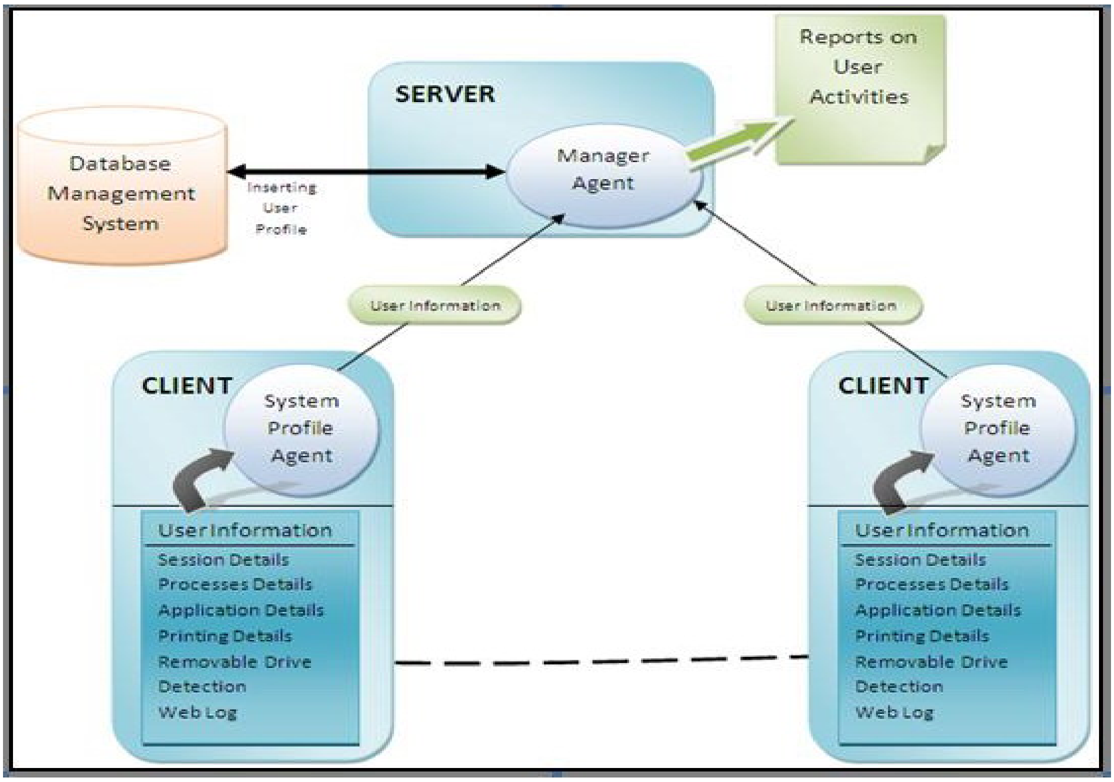

Figure 1. Profile attributes containing activities details

3. Design and Functionality

When user logs into the system, the agents get initiated and start transmitting the data towards server side management agents. Management agents establish placing the data into the database and hence analyzer agent prepares reports available for the web reporting. In fact, overall system functioning is accomplished by means of system profiling, cyber, and alarm agents. They capture and monitor the behavior of the users from all machines, time to time, and transmit towards the management agents either in routine or at any particular event. On the other hand, the manager agents keep on inserting the data into the database for the analysis of it by analyzer agent.

The design follows service-oriented architecture. The design may lead to any maximum limit of internal security assurance and more flexible as it can be scheduled in accordance to any organizational level. The service oriented architecture could be scheduled in accordance to any organizational policies in order to classify the behaviors of its users it also additionally facilitates the alarming functioning to prevent the organization from some harmful activities in short it is the perfect and scalable product that has been developed as research product and could be brought to any limit of implementation within the domain of user profiling environment.

The profile is dynamic in nature that is being updated continuously while monitoring the behavior of an insider. The model is generic and works in most of the organizations. Underlying information and activities are recorded in user-profile to avoid threat.

- Login at unusual times
- Login from unusual locations
- Unusual execution of the processes
- Unauthorized execution of the applications
- Unauthorized attempts to access restricted sites
- Long and short time stay in the organization
- Unauthorized plug-in External Devices such as USB drive in the system
- Idle time of the user in the organization

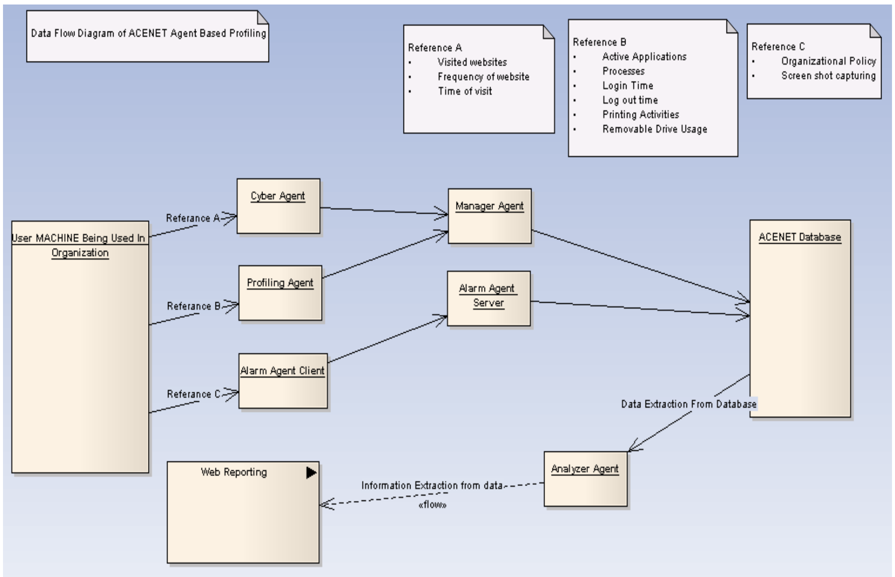

Figure 2 (a). Data Flow Diagram of the Profiling through ACENET€.

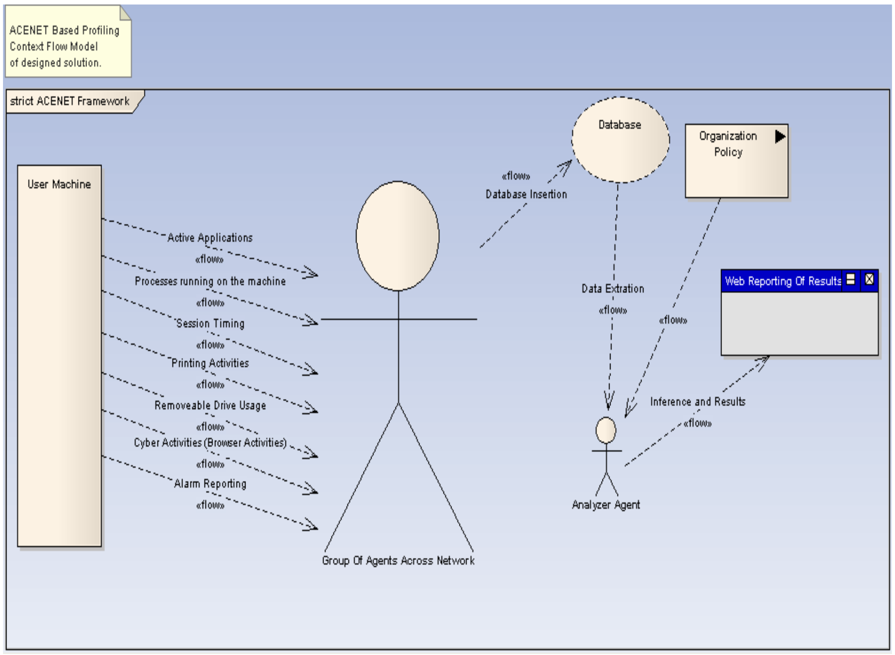

Figure 2 (b). Context Flow Diagram of the Profiling through ACENET.

The next phase of the design will be to develop a complete agent-oriented model where analysis will also be accomplished through agents. Some AI based classifications will be implemented for decision making regarding acceptable and unacceptable profile of the insiders. It will be developed over ACENET.

4. Implementation Details

The second phase of the framework discusses the implementation of profiling over the proposed framework. Followings are some diagrams that illustrate the model to show the flow and the extraction of the information within the framework that leads towards the useful conclusions [10]. Figure 2(a) and 2(b) are showing two different diagrams that explain the interaction of the machine and agents in the model.

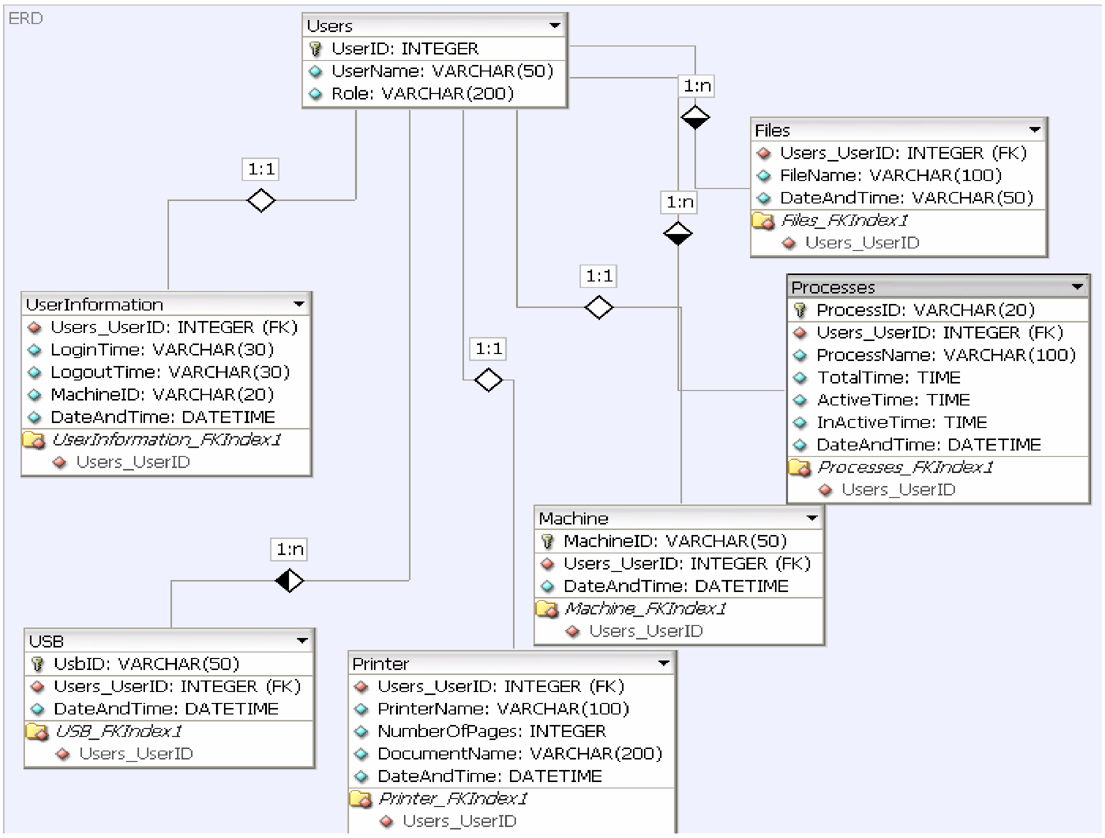

Figure 3. Entity relationship diagram of the framework.

Figure 2(a) shows that the client side agents, i.e. cyber agent, profiling agent and the alarm agent are monitoring behavior, while they are controlled and managed by server side agents, i.e. manager agent and alarm agent.

Figure 2 (b) shows the detailed activities that agents are performing at the client and server side. These activities are stored into the database and then analyzed by the Analyzer agent that checks that in the light of the organization’s policy and finally results are dumped to the web server. The following Entity Relationship Diagram, shown in Figure 3, is very well normalized and flexible that is flexible enough to be used by varieties of the organizations.

Console based coding and implementation has been embedded into the behaviors of the different agents for autonomous execution. All agents were created independent of each other.

5. Conclusion

The developed model generates and updates users’ profiles to avoid and detect insider threat. The Vulnerability Assessment and the profiling model provide foundation to the developed model. In the developed model, agents play pivot role to create and maintain the profiles of the users that are presently applied to detect threat and in future calculate the threat level to avoid it. The model supports distributed environment to solve distributed kind of applications. Manager agents and the Profiling agents have been designed for implementation of the model. Profiling agents record all actions of the authorized user and present all recorded behavior to the Manager agent for further processing. The Manager agent creates and updates profiles of the enterprise users to monitor the behavior as threat may be avoided, detected or recovered. At this moment, the execution of agents has been tested on ACENET whereas the execution will also be tested on other FIPA supporting agents platforms. The agents will also be protected from interruption of users, in future.

Acknowledgement

This is extended version of our own work accepted in International Symposium on Biometrics and Security Technologies, IEEE-ISBAST'08, 23-24 April 2008. The research was conducted at CRUC FAST Karachi campus under guidance of Professor Dr. Zuabir A. Shaikh, Director, FAST-NU Karachi campus.

References

[1] G. Ali, N.A. Shaikh, and Z.A. Shaikh, Towards an Automated Multiagent System to Monitor User Activities Against Insider Threat, Accepted in International Symposium on Biometrics and Security Technologies, IEEE-ISBAST'08, published as Conference Proceedings of ISBAST’08, April 23-24 (2008) p. 1–4.

[2] A. Kamra, E. Bertino and G. Lebanon, Mechanisms for Database Intrusion Detection and Response, Proceedings of the 2nd SIGMOD Ph.D Workshop on Innovative Database Research (2008).

[3] D. Cappelli, A. Moore, T.J. Shimeall and R. Trzeciak, Common Sense Guide to Prevention and Detection of Insider Threats, Carnegie Mellon University, July 2006.

[4] Framingham, CSO Magazine E-Crime Watch Survey (CERT/CC), Sept. 6 (2006).

[5] G. Hinson, The Value of Information Security Awareness, CISSP, CISM, CISA, MBA, CEO, IsecT Ltd. Updated September (2005).

[6] P. Soleimani, R. Noorossana and A. Amiri, Journal of Computers and Industrial Engineering **7** , No. 3 (2009) 443

[7] G. Ali, N.A. Shaikh and A.W. Shaikh, Australian Journal of Basic and Applied Sciences **4**, No. 3 (2010) 442.

[8] G. Ali, N.A. Shaikh, M.A. Shah and Z.A. Shaikh, Australian Journal of Basic and Applied Sciences **4,** No. 5 (2010) 844.

[9] N.A. Shaikh, G. Ali, M.A. Shah and Z.A. Shaikh, Australian Journal of Basic and Applied Sciences **4,** No. 5 (2010) 851.

[10] G. Ali, N.A. Shaikh and Z.A. Shaikh, The Proceedings of the Pakistan Academy of Sciences **47,** No. 2 (2010) 121.