December 1999

A Multi-Agent Architecture for Internet Security

Jess Yuen
City University of Hong Kong

Felix Leung
City University of Hong Kong

Huaiqing Wang
City University of Hong Kong

Stephen Liao
City University of Hong Kong

A Multi-Agent Architecture for Internet Security

Jess H. K. Yuen, Felix S. K. Leung, Huaiqing Wang, Stephen S. Y. Liao,
Department of Information Systems, City University of Hong Kong
{isjess, isfelix, iswang, ishongko}@is.cityu.edu.hk

1. Introduction

Security issues are always discussed while organizations plan the development of their businesses on the web. The continuous growth of the Internet population motivates more businesses to become connected to the web (Segev et al. 1998). However, this also means that information of companies will be exposed to all Internet users. This causes serious threats and therefore hinders the growth of the web-based business.

Many security solutions have been developed, yet existing security systems have some limitations. Many security systems operate centrally and run on a monolithic architecture. Such design, for example, Intrusion detected system (IDS), may cause single point of failure, and hard to scale to the complexities and sophistication of current organization. (Balasubramaniyan et al. 1998). Therefore, using intelligent multi-agent technology, which promises multiplicity, reusability, reactivity and flexibility, can definitely assist improving security on the Internet. The architecture, which is characterized by multiple agent structure, intelligent security mechanism and efficient coordination among different layers, facilitates the protection of the system from Internet threats.

The intelligent agent we define here is a combination of software agent and intelligent system. Intelligent agent should contain the following properties: autonomy, co-operativity, reactivity, pro-activity and mobility (Wang et al. 1997). Since the Internet changes rapidly, we believe that a flexible and reusable intelligent agent technology is favorable for maintaining Internet security. Our multi-agent architecture contains several kinds of intelligent agents. By deploying and coordinating different types of intelligent agents in the architecture, we can combat security threats efficiently. Reusable agents can be customized for particular Internet security issues without great modification. Additionally, the knowledge of agents can be re-configured and shared dynamically to deal with the rapid change of the Internet environment.

In our previous research, multi-agent architecture was designed in various domains such as decision support systems (IADSS) and artificial intelligent systems (APACS) (Wang et al. 1997; Wang 1997). We believe that the similar architecture can also be applied to the Internet security problem domain. Thus, the objective of this paper is to identify and describe the components inside our proposed architecture that provides intelligent and protective security mechanism to a web-based system.

2. Internet Security Threats

Indeed, the Internet is full of vulnerabilities that are easily exploited. For instance, the connection of TCP/IP, the most common protocol suite on the Internet, can be eavesdropped by outsiders. Thus, data can be accessed and manipulated while it is being transmitted on the public network (Bernstein et al. 1996).

Apart from data manipulation, we are also concerned with the problems of system intrusion from the Internet. Intrusions include unauthorized accesses, denial of services, virus infections and malicious program attacks. Since the private network goes public once it is connecting to the Internet, it is possible that someone may attempt to intrude into the corporate network. Even though the intruders do nothing on the system, such activities will waste the system resources and slow down the performance. More seriously, they will damage the data or devices of the system. Intruders can hack into the private network by several means. For examples, they may use some intelligent program to guess the password, or use some techniques like IP spoofing so as to resemble a trusted machine or person to get unauthorized access to the system.

Furthermore, the intruders attack the host by sending viruses. Virus can do many things to attack the host. For example, it wastes the system resources, makes the system runs slowly, damages physical memory or executes underlying commands to destroy the secondary storage. A good example is a famous incident that happened in November 1989. A program, named Morris Worm, attacked thousands of UNIX hosts on the Internet. The program did nothing but made the infected machine running a meaningless loop again and again, and the bug in the program caused the machines to run slower and slower (Oppliger 1997).

Internet security threats such as data manipulations and intrusions do not only impact our computer systems, but also hinder the growth of the commercial usage of the Internet. Therefore, to protect our system, we must take certain actions against such kinds of threats. Throughout this paper, we will demonstrate how a generic architecture assists multi-agents detecting and analyzing the security threats in a web-based system.

3. Multi-agent Architecture at a Glance

Based on our previously established multi-agents architecture such as APACS and IADSS (Wang et al. 1997; Wang 1997), we enhance the architecture for the

multi-agent systems. We split the knowledge broker into communication handler and knowledge manager. As shown in Figure 1, a generic architecture consists of four main layers: Agents, Communication Handler, Knowledge Manager and Information Repository.

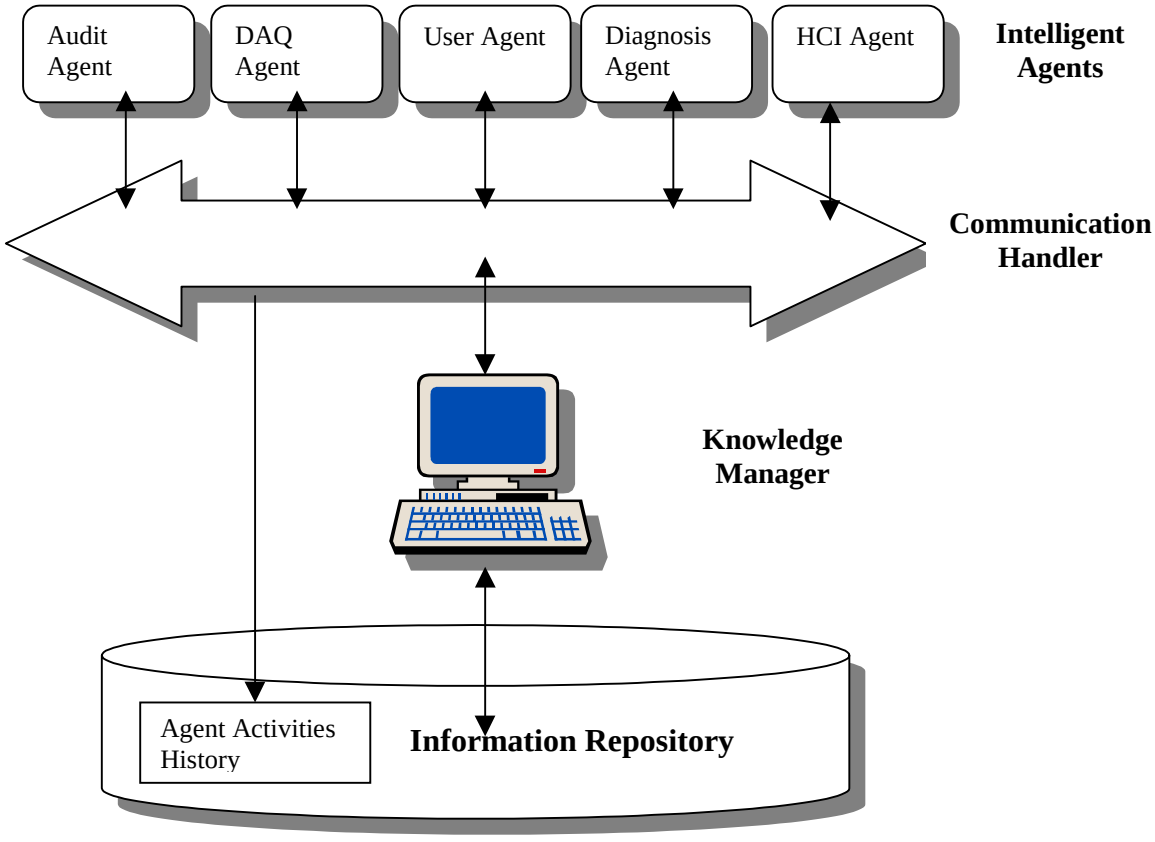

Figure 1 The generic multi-agent architecture for Internet security

Intelligent Agent: Intelligent agent plays various roles in the proposed architecture. There are many kinds of agents such as audit agent, user agent and Data Acquisition agent (DAQ). In a distributed network environment, different agents co-operate together for detecting and analyzing abnormal events that occur continuously or simultaneously. For instance, User agent collects user information from the user. Data Acquisition agent is called for providing appropriate historical data, security rules and operation rules from the information repository. It retrieves data from the information repository via communication handler upon other agents' requisitions. The audit agent checks the system status by comparing and analyzing different sets of data from data acquisition agents and user agents. HCI agent acts as the interfaces between user and the systems for user interactions (Leung et al. 1999).

Communication Handler: The communication handler performs two functions. First, it provides a channel that allows inter-communication among different agents directly. It also avoids frequent access of the knowledge manager which consumes system resources. Second, it provides a common protocol for agents to communicate with the knowledge manager and information repository, and vice versa.

Knowledge Manager: The knowledge manager is an essential component in the architecture. It provides co-ordination control functions over all the agents in the architecture. It sends commands to agents for performing certain operations including creation, termination and services of agents. It filters and classifies the information collected by agents and stores them into the repository accordingly. The manager can also interpret the actions of agents and react intelligently based on its knowledge. Security rules or policy can be set on the knowledge manager dynamically. If there is any abnormal event happens (e.g. certain rule is violated), it will alert to the security administrator immediately and take a corresponding action. Moreover, it manages and co-ordinates the transactions with intelligent agents as provided by the operational facilities components, as well as the synchronization of data from different agents so that significant messages would not be missed.

Information Repository: The repository stores the common knowledge used by all the intelligent agents. It stores on-line information that agents collect from users and systems such as user history and system configurations, management procedures, security rules, operation rules, and agent models etc. It actually provides management for all schema transactions that can be initialized by programs, or be triggered by events, updates and time condition (Wang et al. 1997). In addition, the repository contains a component named Agent Activities History which continuously stores all activities of the agents for backup, further analysis and agent learning.

4. A Scenario

In operation, the DAQ agent continuously receives real time data and sends the data to the knowledge manager. The knowledge manager filters and classifies the data and then distributes them to different agents based on the requests or security rules from repository. It can analyze and interpret the data received and react according to its knowledge.

For example, when a hacker uses a sophisticated program to guess the password to login the system, some specific system variables (e.g. the rate of attempt) will thus be abnormally changed. Therefore, we can set certain security rules in the information repository to monitor such variables. In this case, we can limit the rate of attempt to, for example, 5 per minute. Every time the user attempts to login, a user agent counts the number of attempts and sends to the audit agent (together with other real time data). The audit agent requests the knowledge manager to check the login process. If the attempt rate exceed the limit, the knowledge manager will terminate the connection immediately. Meanwhile, the knowledge manager stores the relevant knowledge such as IP address, login name and event time on the repository. The knowledge can be shared among agents and combine with other information for future use. The knowledge manager can take further actions such as monitoring the next login that uses the same username, and checking the previous history of the user. By doing so, potential security risks can be prevented.

5. Conclusions

In this paper, we have proposed an enhanced multi-agent architecture that provides an autonomous and intelligent protecting mechanism for Internet security to business. Although many mechanisms have been introduced that deal with particular Internet security threats, we still need a generic architecture to deliver those solutions. Based on previous research on multi-agent architecture, we develop some additional features to enhance the design. Particularly, in our proposed architecture, human-like intelligent agents perform various operations and work collaboratively. The communication handler is employed to avoid the overloading of the knowledge manager. We also enhance the functionality of the knowledge manager like giving it a power of filtering the messages from agents. These new features in the architecture can improve the entire performance of ensuring Internet security. Presently, we are implementing a prototype system that applies our generic architecture to Internet security domain and using ObjectStore® as the repository under Java platform. We believe that this multi-agent architecture will be beneficial to protect Internet security, and become a very important basis for Internet-based systems eventually.

References

[1] Balasubramaniyan, J. S., Garcia-Fernandez, J. O., Isacoff, D., Spafford, E., Zamboni, S., “An Architecture for Intrusion detection using Autonomous Agents” COAST Technical Report, Vol. 5, June 1998, pp1-19.

[2] Bernstein, T., Bhimani, A.B., Schultz, E., Siegel, C.A., *Internet Security for Business*, John Wiley & Sons Inc., New York, 1996, Chapter 5, pp.125-170.

[3] Leung F., Yuen J., Liao S., Wang H., “A Conceptual O-O Model for Internet-based Intrusion Detection Agents”, (Submitted to Association for Information System Conferences Aug 1999).

[4] Oppliger, R., "Internet Security: Firewall and Beyond", Communications of the ACM, Vol. 40, No.5, May 1997, pp.92-102.

[5] Segev, A., Porra, J., Roldan, M., "Internet Security and the case of Bank of America", Communication of the ACM, Vol. 41, No.10, Oct 1998, pp.81-87.

[6] Wang, H., “Intelligent Agent–Assisted Decision Support Systems: Integration of Knowledge Discovery, Knowledge Analysis, and Group Decision Support”, Expert System with Applications, Vol.12, No.3, 1997, pp.323-335.

[7] Wang, H., Wang, C., “Intelligent Agents in the Nuclear Industry”, Computer, November 1997, pp.28-34.