Intelligent Multi-Agent Information Security System

N. Kussul [1], A. Shelestov [1], A. Sidorenko [1], S. Skakun [1], Y. Veremeenko [2]
1) Space Research Institute NASU-NSAU, 40 Glushkov Ave 03187 Kiev Ukraine, inform@space.is.kiev.ua, nkussul@dialektika.kiev.ua
2) Physics and Technology Institute, National Technical University of Ukraine "Kiev Polytechnic Institute", 37 Peremogy Ave 03057 Kiev Ukraine, yur@pth.ntu-kpi.kiev.ua

***Abstract**: It is proposed an agent approach for creation of intelligent intrusion detection system. The system allows detecting known type of attacks and anomalies in user activity and computer system behavior. The system includes different types of intelligent agents. The most important one is user agent based on neural network model of user behavior. Proposed approach is verified by experiments in real intranet of Institute of Physics and Technologies of National Technical University of Ukraine "Kiev Polytechnic Institute.*

Keywords: - neural network, multi-agent system, network security system, user behavior model, intrusion detection system

1. INTRODUCTION

At present the urgency of information security issue has increased greatly. Today there is a great number of commercial Intrusion Detection Systems (IDS). The well-known existing IDS representatives are Haystack, MIDAS, ASAX, etc.

Those systems, while contributing pioneer solutions to the security field, possess certain key drawbacks:

1. The probabilities of false positives and false negatives are too high.
2. Detection of previously unknown types of attacks is unlikely.
3. Operation is isolated for the particular host-based IDS, causing the lack of ability for the detection of distributed coordinated attacks against a computer network.

A number of innovative approaches and new models for network security assurance system have been proposed recently [1]. The basic ideas are as follows:

1. Make the IDS more intellectual in terms of attack detection and data processing by means of rule-based networks, neural networks [2], genetic algorithms, human-like immunology systems, variable-sized Markoff chains [3] etc.
2. Make usage of the particular distributed components of the network security systems cooperative. That will enable detection of new, previously unknown types of distributed attacks compromising the computer network as a whole. For such an idea, the multi-agent system model proves to be very promising [4].

2. AGENT APPROACH

The main point about agents is that they are autonomous: capable of acting independently, exhibiting control over their internal state. Thus: an agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors [5]. Among main agent properties are [6]:

- reactivity: provides an ongoing interaction with its environment, and responds to changes that occur in it,
- proactiveness: means goal directed behavior of agent,
- social ability: ability to interact with other agents via some kind of agent-communication language, and perhaps co-operate with others,
- mobility: the ability of an agent to move around an electronic network,
- rationality: agent will act in order to achieve its goals,
- learning/adaption: agents improve performance over time.

The multi-agents system is meant to be the system of interacting agents. They are co-ordinated by general global purpose (the strategy) but autonomous enough to realize their own tasks within the framework of the general strategy (the own tactics).

Importance of transition to agent paradigm is compared with importance of using object-oriented approach [7]. Agent technology can be effectively applied in different areas of information technology, e.g. computer networks, software development, object-oriented programming, artificial intelligence, human-machine interaction etc.

Thus multi-agent model provides significant boost to IDS performance and stability if compared to conventional approaches.

3. SYSTEM STRUCTURE AND FUNCTIONALITY

Integrated network IDS should detect different attack types (known and unknown) and anomaly activities. To meet these requirements it should contain various (autonomous) interactive modules. Such architecture (Fig. 1) can be implemented on the basis of agent approach.

The proposed system contains such agent types:

User Agent. This agent is used to detect anomalies in user activity. The detection of illegal user is carried out on the basis of neural network. We applied on-line multilayer feed forward neural networks. Its aim is to reveal regularities between inputs and outputs. The learning of such type of neural network consist in minimization of error functional by gradient descent method. Weight modification is made according to

$$W(t+1) = W(t) - \eta \frac{\partial E}{\partial W} \qquad (1)$$

where E – error functional, W – weight matrix, η –

learning rate, *t* – step modification.

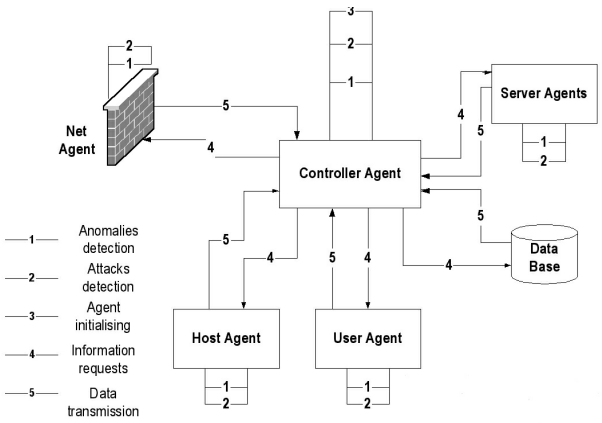

Fig. 1 – System structure.

The goal of functioning of neural network is to predict user processes (running by user) on the basis of user model (formed during neural network learning) and compare them to real activity. The prediction of user processes is made on the basis of 5 previous ones. Also we should take into account that behavior of the same user differs for various operating systems. Consequently, User Agent is developed for each type of operating system available in the network (e.g. Win2000, 98, XP, Free BSD).

For input data binary coding was applied. We used 7 bits for each process, thus the dimension of input data space is 35. In turn, for output data decimal coding was applied, and the dimension of output data space is 1.

As to neural network architecture, we are using neural network with 3 layers:

- input layer with 35 neurons,
- hidden layer with 10 neurons, and
- output layer with 1 neuron.

Data for neural network learning were obtained during a real work of users in Space Research Institute NASU-NSAU and National Technical University of Ukraine "Kiev Polytechnical Institute". For this purpose special developed by us logging software was applied. Hence, train and test data were formed in such way:

- first, log files were processed in format suitable for neural network;
- then data were mixed and divided into train and test sampling.

Train and test sampling contained about 1000 vectors (70 % for train and 30 % for test data). Indexes of predicted processes for train and test data, along with indexes for illegal users shown in Fig. 2.

1 - Index of predicted processes for legal user on training set.

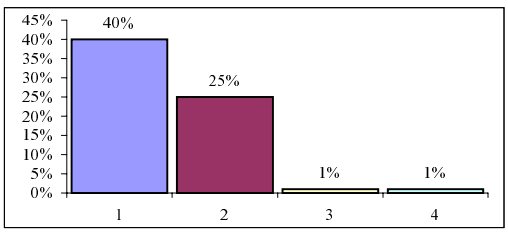

Fig. 2 – Indexes of predicted processes for different users.

2 - Index of predicted processes for legal user on testing set.

3 - Index of predicted processes for illegal user #1.

4 - Index of predicted processes for illegal user #2.

Fig. 3 shows the ability of network to distinguish illegal user under legal account.

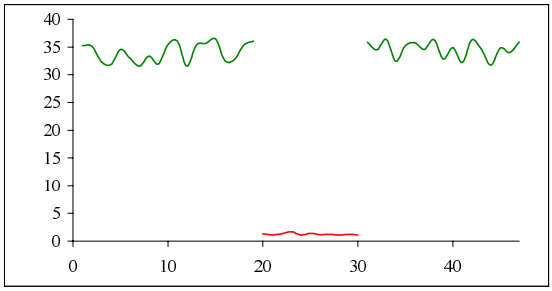

Fig. 3 – Index of predicted processes (green line – for legal user, red line – for user working under another's account).

Thus, these results show the possibility of neural network to detect anomalies in user activity.

Host Agent. Performs system calls processing and detects anomalies and known types of attacks. For example, it allows detecting "Trojan horse" attacks.

Network Agent. Operates at the firewall and analyses the network traffic. The information extracted from packets is used to detect known attacks and anomalies in the network. It may be done utilizing neural networks and probability approaches (e.g. Bayesian networks and variable-sized Markov chains).

Server Agents. Agents that are responsible for the server security.

Controller Agent. Responsible for anomalies analysis and detection of distributed attacks in scale of whole system, initializing and co-ordination agents, interaction with database and between various parts of the system.

Database. Contains data for different agents.

As the user logs on, Controller Agent creates correspondent User Agent and initializes it. At the same time User Agent gets data about user model. During the user's session agent controls the user's activity on the basis of neural network behavior model. At the same time it picks data for behavior model correction. When the session is finished it sends data for database update. In the case of anomaly detection User Agent informs Controller Agent about suspicious activity.

Host Agents and Server Agents detect system anomalies and known attacks.

4. REALIZATION

For realization of proposed system and agents Java and Aglets Software Development Kit (ASDK) were chosen. Java as a programming language for agents offers the set of unique features for multi-agent systems realization. ASDK is free-ware provided by IBM. Aglet API defines fundamental functionality of agents and contains following principle interfaces and classes:

- AgletProxy
- AgletContext
- Message

Aglets exist only within AgletContext. The access to

the agent is possible only through AgletProxy that provides one of the mechanisms for aglets security.

According to system functionality the following classes were designed:

- HostAgent
- UserAgent
- DBBroker
- Watcher
- Controller

Since aglets exist only within AgletContext the application was developed to provide the platform for aglets existence according to the following scheme [8]:

- platform parameters setting
- initialisation of AgletRuntime instance
- user authentication (context owner)
- creating MAFAgentSystem_AgletsImpl instance
- factory components installation
- creating AgletContext instance
- creating ContextListener instance and adding it to the created context
- security manager installation
- context start
- communication layer start

The further development of the system functionality lies in programming agents' behavior – its interaction and messaging. Also some new methods and attributes were added to the Aglet class.

5. CONCLUSION

The above approach takes advantage of both intellectual methods of intrusion and anomaly detection and multi-agent architecture. The use of neural networks enables detection previously unknown attacks types, while agent-based architecture provides features of intelligence and scalability as well as possibility to work in a heterogeneous environment. Currently, research of user behavior model demonstrates effectiveness of such approach.

6. REFERENCES

[1] V. Gorodetski, O. Karsaev, A. Khabalov, I. Kotenko, L. Popyack, V. Skormin. Agent-based model of Computer Network Security System: A Case Study. *Proceedings of the International Workshop "Mathematical Methods, Models and Architectures for Computer Network Security"*, Lecture Notes in Computer Science, vol. 2052, Springer Verlag, 2001, pp. 39-50.

[2] J. Cannady, J. Mahaffey. *The Application of Artificial Neural Networks to Misuse Detection: Initial Results*.

[3] A. M. Sokolov. Computer System Intrusion Detection utilizing second-order Markoff chain. *Artificial Intelligence*. Vol. 1, pp. 376-380. (in Russian).

[4] J. S. Balasubramaniyan, J. O. Garcia-Fernandez, D.Isacoff, E. Spafford, D. Zamboni. An Architecture for Intrusion Detection using Autonomous Agents. *http://citeseer.nj.nec.com/balasubramaniyan98architecture.html*.

[5] S. Russel, P. Norvig. *Artificial Intelligence: A Modern Approach*. Upper Saddle River NJ: Prentice Hall, 1995.

[6] M. Wooldridge. *An Introduction to Multiagent Systems*. Chichester, England: John Wiley & Sons, 2002.

[7] M. Luck, P. McBurney, C. Preist, C. Guilfoyle. Agent Technology Roadmap. AgentLink 2002. *http://www.AgentLink.org*.

[8] M. Oshima, G. Karjoth. Aglets Specification (1.0). IBM Tokio Research Laboratory, May 1997. Version 0.60. *http://www.trl.ibm.com/aglets/spec10.htm*.