An Autonomic Intrusion Detection Model with Multi-Attribute Auction Mechanism

Qingtao Wu, Xulong Zhang, Ruijuan Zheng and Mingchuan Zhang

Electronic & Information Engineering College, Henan University of Science and Technology
Luoyang, Henan Province 471023, China

Abstract

We present an innovative intrusion detection model based on autonomic computing to extend the passive detection mechanism in a traditional intrusion detection system (IDS). Centered on an autonomic manager, this model introduces a multi-attribute auction mechanism in the agent coordination layer to perceive environmental changes, manage and allocate resources, and achieve an active response to intrusions or attacks. Experimental results show that the model can improve the adaptability and detection accuracy of the IDS effectively, through its rational parameter configuration capability.

Keywords: *Intrusion detection, Autonomic computing, Auction mechanism, Agent coordination.*

1. Introduction

Intrusion detection is a widely used and important network security technology that can improve safety greatly and reduce security threats to a system by creating a dynamic safety cycle. With the development of large-scale networks and the establishment of complex requirements involving network intrusion, there are now many demands on intrusion detection technology. Existing intrusion detection systems (IDSs) can offer only passive detection mechanism, wherein, only when an intrusion or attack has occurred can the IDS respond. An intrusion or attack may therefore still cause local or widespread compromises to system safety. In essence, IDS is a post-mortem mechanism that can identify an event only after it has already occurred. It can report the event, but has no adaptive ability. Artificial intelligence, mobile agents, data fusion, information correlation [1–3], and other technologies and methods have been introduced by researchers into continuous intrusion detection, aiming to identify an attack in a timely and effective manner.

Autonomic computing can overcome the heterogeneity and complexity of computing system, has been regarded as a novel and effective approach to implementing autonomous systems to address system security issues. The “autonomic” is inspired by the autonomic nervous system of the human body, which can manage several key functions via involuntary control. Autonomic system is the adjustment of the software and hardware resources of a system to manage its operation, driven by changes in internal and external demands. It has four main characteristics, namely self-configuration, self-healing, self-optimization, and self-protection. The core of an autonomic system enables the computer system to realize high reliability, availability, and service performance.

However, studies of security technology based on autonomic computing have focused only on safety technology. In this case, action is delayed until after the system is attacked, with system safety being compromised and the intrusion not being detected in time. For the purposes of system safety, an autonomous system combined with intrusion detection technology that enables dynamic adaptation to environmental changes, thereby achieving a timely detection of intrusion, should be investigated. The present study combines intrusion detection with autonomic computing to improve the poor adaptive capability in the passive detection mechanisms of traditional intrusion detection technology. To achieve this, we propose an autonomic characteristic intrusion detection model (ACIDM) with auction mechanism.

2. Autonomic intrusion detection model

The proposed autonomic intrusion detection model is shown in Fig. 1. In this model, the managed resource (MR) [4] covers all types of physical and virtual resources, such as databases, servers, routers, application modules, Web servers, virtual machines, host logs, network packets, and firewall alarm messages. These resources must be manageable, observable, and adjustable. The state of the resources refers to all data (events) reflecting the existing resource state, including log and real-time events, such as the operative and performance status (throughput and availability of resources) of the resources, and anomalous events. The MR is uniformly distributed and managed by an agent coordination layer.

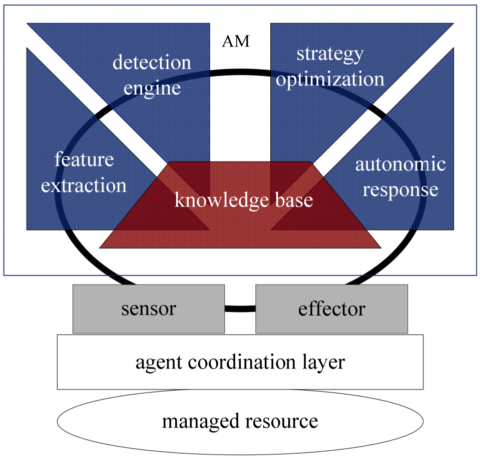

Fig. 1 Autonomic intrusion detection model

The autonomic manager (AM) in Fig. 1 consists of feature extraction, detection engine, strategy optimization, autonomic response, and knowledge base. The line connecting these four parts indicates the sharing information and message. Strategy optimization requires the detection engine to collect additional information before operating. Coordination among these four elements can be implemented via asynchronous communication.

2.1 Agent Coordination Layer

The agent coordination layer uses different intelligent agents for the various MRs to provide data support for the AM. These agents are entities that can operate independently. Agents capture MR information and remove redundancy by preprocessing before its final submission to the AM. Another major function of the agent coordination layer is to receive feedback regarding AM information and adjust the system environment autonomously to adapt to changes. In addition, the agent coordination layer realizes the dynamic configuration of resources, the synthesis of services, and the calibration of system parameters. For example, when the system detects intrusion, the agent controlling the firewall will update the blocking strategy based on the intrusion alarm and will control information from the AM to block subsequent attacks over a certain time according to IP address, interface, and other information. This process can be described as IDS with dynamic self-adaption.

The agents in the agent coordination layer work synergistically to form a multi-agents system. An auction mechanism is introduced by the multiagent system to resolve task allocation, resource configuration, and system performance optimization [5–6]. A variety of auction methods serve the different environments. The multiattribute auction method defined below was used in the present paper.

Definition 1. Multiattribute auction model
In this model, $M = \langle A, B, S, V, C, \mathrm{Re}s \rangle$, where A refers to attribute space and $A = A_1 \times \cdots \times A_m$. Every auctioned event includes the m attributes (i.e., $a_1, \cdots, a_m$). The value range is $A_1, \cdots, A_m$, and we specify $a = (a_1, \cdots, a_m)$ as an attribute vector of the event, so $a \in A$.

B refers to a unique buyer at auction who needs to purchase an event. S refers to a seller set, which includes n buyers, i.e., $S = \{s_1, \cdots, s_n\}$. The buyers can provide events with different attributes.

$V : A \to R$ refers to the attribute assessment function of buyer B (R is the set of real numbers). The assessment value that buyer B made for an event with an attribute of a is $V(a) \in R$.

In this model, $C = \{C_1, \cdots, C_n\}$, where C_i refers to the cost function for seller i. The amount that seller i receives for an article with an attribute of a is $C_i(a) \in R$.

$\mathrm{Re}s$ refers to a transaction program. $\mathrm{Re}s = (P, a)$, where the knockdown price is $P \in R$ and the transaction attribute vector is $a \in A$. At this time, the benefit of the buyer B is $U = V(a) - P$. The benefit of the seller S_i is $U_i = P - C_i(a)$.

The process of the auction is divided into four steps:

(1) The buyer publishes an evaluation function $V^{'}(a)$ ($V^{'}$ may differ from V).

(2) Each seller i makes a sealed bid $B_i \geq 0$.

(3) The transaction seller is confirmed. First, the buyer decides on an optional seller set $W = \{\omega \mid (\omega \in S) \wedge (B_\omega = \max_{i \in S}(B_i)) \wedge (B_\omega > 0)\}$

where B_ω refers to a bid for ω. If $W = \varnothing$, no transaction seller exists, and the auction ends. If $W \neq \varnothing$, $\omega \in W$ is generated randomly as a transaction seller. We define

$$B^* = \max_{i \in S}{}_2(B_i)$$

where $\max_{i \in S}{}_2(B_i) \underline{\underline{def}} \min_{i \in S}(\max_{j \in S-\{i\}}(B_j))$. Here, $\max_2$ is the maximal value of the residual element after removing the maximal element (i.e., $\max_2(1,2,3) = 2$, $\max_2(1,2,3,3) = 3$). The quantity B^* is then the highest bid of other sellers, except for the seller ω.

(4) The transaction process is proposed by the transaction seller (P_t, a_t) . The legal proposal should satisfy $V^{'}(a_t) - P_t = B^*$, based upon which the transaction seller reaches a deal with the buyer. The auction then ends.

2.2 Sensor and Effector

The hardware and software of the distributed system may be sourced from different service suppliers. A standard interface is therefore needed to prevent the heterogeneity with respect to the resources. The fundamental method for resolving this issue is to establish a sensor and an effector through standardization and semantic technology. Based on the theory related to the sensor, a formalized definition can be obtained, as follows.

Definition 2. Sensor
Let $T = \{t_1, ..., t_n\}$ be a group of feature sets that can reflect the existing MR state and let $V = \{v_1, ..., v_m\}$ be a group of event sets that reflects an MR state change. Let $O = (C, R)$ be the domain ontology, where C is the domain concept set and R is related to C . Let $\xi = \{get, report\}$ be a group of operation sets. $Sensor$ can then be defined as a 4-tuple: $Sensor = (T, V, O, \xi)$, where $\forall t_i, v_j (1 \le i \le n, 1 \le j \le m)$, $t_i \in C, v_j \in C$.

The sensor supports automatic interpretation and reasoning and realizes self-awareness. The MR feature and event sets comply with the specific domain ontology expressed in an ontology language with clear semantics. The operation get was used to capture MR state features, with $get(t_i)$ indicating that the AM obtains the characteristic t_i from MR via the sensor. The operation $report$ was used for reporting MR state changes, with $report(v_j)$ indicating that the MR reports v_j to the AM.

Definition 3. Effector
Let $A = \{a_1, ..., a_n\}$ be a group of executable action sets that can operate the MR state and let $Q = \{q_1, ..., q_m\}$ be a group of action sets released by the AM for MR application. Let $O = (C, R)$ be the domain ontology, where C is the domain concept set and R is related to C . In this study, $\psi = \{set, request\}$ is a group of operation sets. $Effector$ can then be defined as a 4-tuple: $Effector = (A, Q, O, \psi)$, where $\forall a_i, q_j (1 \le i \le n, 1 \le j \le m)$, $a_i \in C, q_j \in C$.

In Definition 3, the operation set was used for execution action, with $set(a_i)$ indicating that the AM executes action a_i through the effector. The operation $request$ triggers the MR to send a request (e.g., for help or consultation) to the AM, with $request(q_j)$ indicating that the MR executes the request action q_j to the AM.

2.3 Data Normalization

The collected data should be preprocessed to resolve heterogeneity. Normalization theory [7] is adopted to unify the type and format of the data. In an IDS, the Euclidean distance between characteristic vectors must be calculated. This distance should normalize the process, because leading one numeric data item to affect another is easy for the sake of the difference in value ranges. The steps of the processing method are as follows. First, the mean and standard deviation for training each characteristic attribute of sample are calculated as follows.

$$mean[j] = \frac{1}{n}\sum_{i=1}^{n} ins\tan ce_i[j] \qquad (1)$$

$$s\tan dard[j] = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(ins\tan ce_i[j] - mean[j])^2} \qquad (2)$$

where $ins\tan ce_i[j]$ is attribute j in the training sample i , and n is the number of samples. This sample from the training collection is transferred as follows.

$$newins\tan ce[j] = \frac{ins\tan ce[j] - mean[j]}{s\tan dard[j]} \qquad (3)$$

Formula (3) can be used to transfer the value of the attribute to multiple standard deviations. Considering that this value deviates from the mean, it can map the attribute value of a sample from its value space to a standard value space.

2.4 The AM

Knowledge base: “Knowledge” refers mainly to state determination (KD), strategy knowledge (KP), problem solving knowledge (KS), and detection rules (KR). That is, the knowledge base is K=KD+KP+KS+KR. In this equation, KD is used mainly to monitor the state parameters of the managed resources and the internal and external environments. KP mainly includes the strategy defined by the IT manager and the strategy obtained through machine learning (i.e., mapping from state to action). KS mainly includes rules, configurations, and optimizations, and how to solve problems when the running state of the system deviates from expectations. KR mainly includes the characteristic base derived from the misuse of intrusion detection, and the behavioral model base derived from abnormal intrusion detection. Other subsystems run with the support of the knowledge base.

Feature extraction: The sensor obtains the data captured by the agent coordination layer. Expansion matrix theory is used to extract the intrusion characteristics [8] through analysis, relationships, and data integration. This method establishes an integer programming model selected by its optimal characteristic subset through the creation of an expansion matrix of intrusion and normal subsets. In addition, this method can generate an optimal rule for detecting a specific type of attack using a simple genetic algorithm.

Detection engine: The detection engine is a functional component, performing detection for the AM. It can identify the intrusion intention using mixed detection technology.

Strategy optimization: Strategy optimization is realized by adopting machine learning, intelligent planning, and other related technologies that can adapt to environmental change.

Autonomic response: Autonomic response completes the response to intrusion according to the strategy knowledge in the database.

3. Simulation Experiments and Performance Analysis

3.1 Experimental Data Set and Design

We adopted the KDD Cup 1999 data set [7], which has been approved and adopted widely in the intrusion detection research field as a benchmark for detection, to validate the experiment. This data set includes approximately 4,900,000 data records. The records were extracted from original network data obtained by a simulated attack on a military network environment. The data are based on a set of 41 characteristic vectors describing statistical information about network connections that include five kinds of data. Among these data types are four kinds of attack data (namely Dos, Probe, R2L, and U2R, with 24 kinds of attachment types in total) and one type of normal data. The 41 characteristics of this data set are mainly categorized into two data types: numerals and nouns. The numeric data are processed first. The noun attributes in the data set, including protocol and service types, are processed using data normalization based on the occurrence frequency of each value in the value range. Therefore, the value of an attribute ranges from 0 to 1. In the current experiment, 10% of the selected data set was used as experimental data.

The following two experiments were designed to investigate the feasibility and effectiveness of the proposed model:

Experiment 1: A comparison of the performances of AM detection engines with respect to detection accuracy, using mixed and misuse detection technologies.

Experiment 2: A comparison of the performances of ACIDM and two intrusion detection models with respect to detection accuracy and time. The two other models were an artificial neural network (ANN) and a support vector machine (SVM).

3.2 Experimental Results and Analysis

First, the same data set was adopted for the two experiments. The comparison of AM detection engine accuracies using mixed and misuse detection technologies is shown in Table 1.

Table 1: Comparison of misuse and mixed detection technology performances

ttack method	Detection rate (%)	
	Misuse detection	Mixed detection
ormal	93.26	98.35
Dos	83.47	98.64
U2R	76.68	95.28
R2L	74.57	94.75
Probe	86.69	98.43

Table 1 shows that the detection performance using a mixed detection technology is significantly better than that using misuse detection technology.

The detection accuracy and time performance of the ACIDM were compared with those for the ANN and SVM intrusion detection models. The experimental results are shown in Table 2 and Fig. 2.

Table 2: Detection accuracy of ACIDM, ANN, and SVM

Attack method	Detection rate (%)		
	ANN	SVM	ACIDM
Normal	82.21	93.26	98.35
Dos	67.35	83.47	98.64
U2R	64.28	76.68	95.28
R2L	69.57	74.57	94.75
Probe	76.24	86.69	98.43

The ACIDM performed substantially better with respect to detection accuracy than did the ANN and SVM models, as shown in Table 2. The performance comparison for detection time values is shown in Fig. 2.

Table 2 shows that the ACIDM performs better than the ANN and SVM models in terms of detection accuracy. However, the ACIDM fared poorly with respect to detection time because it adopts a mixed intrusion detection technology in the detection process and adds autonomic-response and response-strategy optimization to improve the self-adaptability of the system, thereby extending the detection time. The detection time for the SVM model was the shortest.

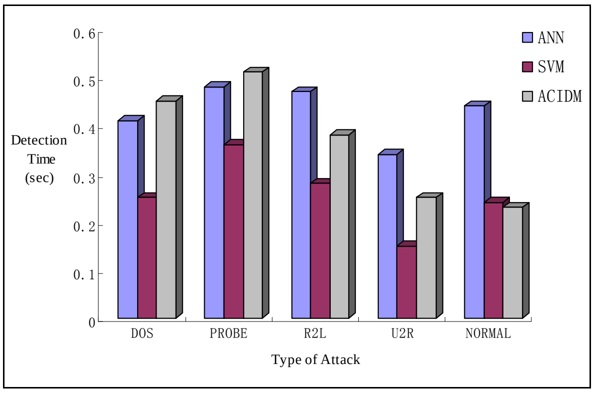

Fig. 2 Detection times for ACIDM, ANN, and SVM

The following conclusions can be derived from these experimental results:

(1) The detection engine should adopt a mixed detection technology. Its detection accuracy was considerably better than that for the misuse detection technology.

(2) The ACIDM performs better than ANN and SVM in terms of detection accuracy. However, its detection time is greater.

4. Conclusions

Focusing on resolving the drawbacks of the passive detection mechanisms in traditional IDSs, an autonomic intrusion detection model with auction mechanism was proposed in this paper. The model is centered on the AM and integrates a multiattribute auction mechanism, which can perceive changes in the system environment, into the agent coordination layer. This can perceive changes in the system environment, and adapt the configuration management accordingly. The experimental results show that the model can enhance the self-adaptive performance of a system and obtain high detection accuracy with appropriate settings. Although the ACIDM demonstrated a high detection performance, the detection time was relatively long, which should be the focus in further research.

Acknowledgments

The authors thank the anonymous reviewers for their valuable comments and suggestions. This work is sponsored partially by the National Natural Science Foundation of China (No. 61003035) and the Plan for Scientific Innovation Talent of Henan Province (No. 124100510006)

References

[1] Y.Y. Zhang, W. Nurbol, J.Q. Cheng-ming, and L. Hu. "Status of Intrusion Tolerance", Journal of Jilin University (Information Science Edition), Vol. 27, No. 4, 2009, pp. 389–394.

[2] Shakhatreh. Ala' Yaseen Ibrahim, and Bakar. Kamalrulnizam Abu, "A Review of clustering techniques based on machine learning approach in intrusion detection systems", International Journal of Computer Science Issues, Vol. 8, No. 2, 2011, pp. 373-381.

[3] Qingtao Wu, Ruijuan Zheng, Guanfeng Li, Juwei Zhang. "Intrusion Intention Identification Methods Based on Dynamic Bayesian Networks", Procedia Engineering, Vol.15, 2011, pp.3433-3438.

[4] LI Bing-yang, WANG Hui-qiang, FENG Guang-sheng, "Model construction and quantitative analysis of autonomic intrusion tolerance system", Application Research of Computers. Vol. 26, No.5, 2009, pp. 1883-1887.

[5] D. H. Shih, D. C. Yen, C. H. Cheng and M. H. Shih. "A secure multi-item e-Auction mechanism with bid privacy", Computers & Security, Vol.30, No.4, 2011, pp.273-287.

[6] Noman Mohammed, Hadi Otrok, Lingyu Wang, Mourad Debbabi, Prabir Bhattacharya. "Mechanism Design-Based Secure Leader Election Model for Intrusion Detection in MANET", IEEE Transactions on Dependable and Secure Computing, Vol. 8, No. 1, 2011, pp. 89-103.

[7] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani. "A Detailed Analysis of the KDD CUP 99 Data Set", Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), 2009, pp.1-7

Qingtao Wu got his PhD degree in computer science from East China University of Science and Technology, in 2006, on network and information security. He's Associate Professor in Computer Science at Electronic & Information Engineering College of Henan University of Science and Technology, China. He's currently managing and leading 2 projects supported by the National Natural Science Foundation of China to address the autonomic mechanism for the retainment and enhancement of system security. His main research interests include computer system security, intelligent information processing, etc.

Xulong Zhang received his Bachelor's degree in Computer Science and Technology in 2011. He is currently a Master Degree

Candidate directed by Dr. Qingtao Wu in Computer Science at Electronic & Information Engineering College of Henan University of Science and Technology, China. His research is focused on network security.

Ruijuan Zheng got her PhD degree in computer science from Harbin Engineering University, in 2008, on autonomic system security. She's Associate Professor in Computer Science at Electronic & Information Engineering College of Henan University of Science and Technology, China. Her main research interests include computer system security, network security, etc.

Mingchuan Zhang got his master degree in computer science from Harbin Engineering University, in 2005, on intelligent information processing. He's lecturer in Computer Science at Electronic & Information Engineering College of Henan University of Science and Technology, China. Her main research interests include computer system security, intelligent information processing, etc.