An Intrusion Detection Method Study Under the Environment of IPv6

Wang Chao
College of Information Science & Electronic Technology
Jiamusi university
Jiamusi,154007,China;
e-mail: shuzi0920@163.com

Wang Bin*(Author for correspondence)
College of Information Science & Electronic Technology
Jiamusi university
Jiamusi,154007,China;
E-mail:jmsuwang@163.com

Zong-li Zhang
College of International Students Education
Jiamusi University
Jiamusi,154007,China;
E-mail: jeanneettee@163.com

Abstract—**The intrusion detection under the environment of IPv6 is an important security technology along with firewall in system security defense system, which can be used for real-time detection and monitoring of the system in the whole process of system invasion. This paper puts forward an intrusion detection system under IPv6 platform based on intrusion detection feature attribute reduction by using pattern matching, so as to expand the range of application and user group of the security products. By the analysis and comparison of various pattern matching algorithms, the new algorithm realizes the intrusion feature module matching under IPv6, and make detection system be of high efficiency. Later experiments have proved this view.**

Keywords- IPv6; pattern matching, intrusion detection

I. INTRODUCTION

With wider and wider IPv6 application, the network security problem under the environment has become a common challenge confronting human in information age and our domestic IPv6 network security issues are becoming increasingly prominent. Concrete manifestations are: 1. The situation of computer system's virus infection and damage is quite serious [1 ~ 3]. According to a recent survey [1], the Chinese computer users' computer virus infection proportion is higher under IPv6 environment; about 73% of the computer users have virus infections. Among them, the users infected more than three times reaches as high as 59%, and the virus have bigger destructiveness. All data damaged by virus accounts for 14%, and part loss occupies 57%. All kinds of virus variants occur extensively in the Internet under IPv6 environment. The viruses are transmitted through various means, and alter all the icons of program files affected to a panda holding three incenses, at the same time the viruses can also steal the users' game account number, and QQ account, etc. This would cause great harm to the users, and the direct and indirect economic losses are inestimable [4 ~ 5].

II. INTRUSION DETECTION MODEL UNDER IPV6 ENVIRONMENT BASED ON THE USE OF PATTERN MATCHING

A. Intrusion detection feature attribute reduction

Because the design of Snort open source and its intrusion detection rules is good, Snort design rules are adopted as our rule system in our system design process. According to the data source of detection, the intrusion detection system can be divided into host-based intrusion detection system and network-based intrusion detection system. Host-based intrusion detection system is to find possible invasion through the analysis of the audit data and system logs. Network-based intrusion detection system is to detect possible invasion through the analysis of network packets.

Snort is a network-based intrusion detection system, which is to detect invasion through misuse detection rules. Snort rule is to use "a simple, lightweight description language" to describe the data package with attack logo in the network. Snort rules file is the core of Snort, and is the attack knowledge base of Snort. If only with Snort executable program and no rules file, then Snort cannot truly achieve intrusion detection function. That is it can't identify any attack. Snort rules are logically divided into two parts: Rule Header and Rule Option.

Rule header is composed of 3 parts: rules behavior, protocol field, address and port information.

a) Rules behavior:

Snort defines five kinds of optional behaviors: alert, log, pass, activate, and dynamic. The semantics are as follows:

Alert: use the set warning methods to generate warning information, and record this message.

Log: use the set record method to record this message.

Pass: ignore this message.

Activate: proceed alert, and then activate another dynamic rules.

Dynamic: wait to be activated by a activate rule and proceed logo.

b) Protocol field

The current Snort supports IP, TCP, UDP and ICMP and may support more agreements in the future.

c) Address and port information

Format: IP/CIDR Port

The CIDR block denotes network mask, such as 192.168.1.0/24. 24 refer to 24 bits, equivalent to mask 255.255.255.0. Address part can take a single address or address list; the same port part can take a single port, port range or port list; also can use negative operator!; and arbitrary operator any.

Rule option part contains warning information showing to users and other messages used to determine whether it is the attacking message (such as tcp's flag fields, the content of the data field, etc.).

B. *Intrusion detector based on concept lattice*

Network intrusion detection process based on the rule matching is as follows: rules are organized into a rule set based on attribute classifications like source IP, target IP, source port range and target port range. So, when a packet is detected, Snort detects the above four parameters of each set of rules to determine whether to detect this rule set or transfer to the next rule set. If the packet matches the four parameters of rule set, the intensive rules of this rule get detection in turn, and the rest parameters of each rule are also detected in sequence. When all rules concentrated have been detected, then search the four parameters of next rule, and detection process restarts. Each packet must proceed the matching process from the beginning to the end.

So, intrusion detection is the matching process of packet captured from the network and rule set. If there is a matching rule of the packet, it means detecting an attack. Then process according to the rules of designated actions (such as warning, etc.). If the searching of all rule sets did not find the matching rule, the packet is normal.

Usually, when rules are few, the detection method is very efficient. However, when rule sets have a lot of rules, standard test method will become very inefficient. Rule detection speed would be slow since each packet must test each parameter of every rule. However, according to the analysis of rule set's structure and matching process, we can conclude that many matching processes are not necessary, and there is large room to be optimized. In addition, with the current increasing network bandwidth, IDS will become a bottleneck of the network bandwidth, and the rule set optimization of intrusion detection system would provide more opportunities for intrusion detection system.

C. *Rules' optimization*

In order to improve the processing speed of detection rules, detection method based on the rule sets need to be applied. The rules optimizer must operate rule sets' construction and selection.

1) *The rule sets' construction*

It is necessary to construct rule sets by use rules optimizer for detection method based on rule sets. Rules optimizer must meet two requirements: (1) construct the smallest and the most efficient rules set impossible; (2) structure discretized rules set. In this way, each packet only searches one rule set.

In the initial stage, the rules optimizer uses the most independent Snort rule parameters to construct rule sets. For each type of transport protocols has different parameters to make it independent, so the rule parameters selected for different transmission protocol are different. Such as: TCP rule set can be differentiated from other TCP rule sets according to the source port and target port, and ICMP rule set can be distinguished according to the ICMP type of the rules. Rules optimizer utilizes independent parameters structure rules subset. This makes the rule sets of many rules detection engine testing smaller. More importantly, it allows respective let through the corresponding rule subset according to the characteristics of packets.

2) *The rule sets' selection*

When Snort operates, rules optimizer selects a rule set for each packet. The rule set choice depends on the matching results of some parameters of the packets received and parameters of rule set. So, rules optimizer only chooses those rules which can match the packet rules and only filter rules that multi-rules search engine need to deal with. Since then multi-rules search engine can further detect content based on rule set detecting methods. For some abnormal data packet, it is possible to select two rule sets. This situation is called "independent conflict".

3) *practical application of rules optimizer*

According to independent parameters of transfer protocol, rules optimizer divides the rules into defined and small rule sets in order to improve Snort detection speed. By analysis, the source port, and target port can be used as TCP/UDP packet independent parameters; ICMP type can be used as ICMP packets independent parameters; Transfer protocol ID can be used as IP independent parameters. Respective introduction are given below:

(a) TCP/UDP

The most independent attributes in TCP/UDP protocol are source port and target port [35]. Normally port has two types: reserved port number and non-reserved port number. This means that most communication between host computers has a reserved port (usually the server) and the other is non-reserved port (more than 1024 and is usually a client). This property allows rules optimizer use reserved port as an independent parameter. As we all know, TCP/UDP between client and server is two-way communication. That is, Snort detection packets are from the client to the server and from the server to the customer. So not every rule is applied to every packet in customer communication and server communication. The rules optimizer groups rules according to whether the independent port is located in the source port or is located in the target port. If independent port is located in the source port, usually data stream is from the server, the server response rules are selected. And when the independent port is located in the target port, usually data stream is from the client, the customer request rules are selected. If the source port and target port are both reserved port, which is less appeared, general rules are applied for its treatment. As is shown in Figure 2:

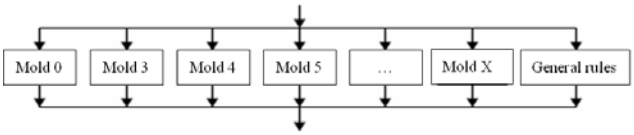

Figure 2. The refining of TCP/UDP rules

However, when the source port and target port are both reserved ports, multiple rule sets need to be detected, and independent conflict appears. Independent conflict can be divided into two types: defined independent conflict and undefined independent conflict. Defined independent rules

are very normal, while undefined independent conflicts are rare, and when it happens, malicious or abnormal events may come along.

For the detection of defined independent conflict, it can be dealt with by a defined independent conflict rule set combining two independent legal rule sets. An example of legal independent conflict is in the DNS query between servers. At this time, two port numbers are 53, and which rule set should be applied to deal with the packet? The answer should be two. Under this circumstance, legal independence conflict should be defined through the agreements. Merge DNS query rules of the source port and target port which are both 53, and add it to the defined independent conflict rules.

When undefined independent conflict occurs, multiple rule sets also need to be detected. It is impossible to follow the defined independent conflict processing to handle all undefined independent conflicts because address space is too big. For example, if there are 1000 independent source ports and 1000 independent destination address rule sets, so there will be 1000000 independent conflict port rule set. Considering the memory needed by a rule set, this is not realistic. Now, depending on user need performance, there are three processing methods for undefined independent conflict: double detection, first hit detection and random testing.

○ 1 Double detection

Double detection is the most basic type in processing undefined independent conflicts. Now that there are two rule sets corresponding to one undefined independent conflict, double detection is to check two rule sets, which would spend about two times of the testing time. Double detection ensures all events of a packet, but this method more easily causes undefined independent conflict packets through the construction to forced double check and cause denial of service (DoS) attack.

○ 2 First hit detection

First hit detection method is similar to double detection method, because it also checks two rule sets. What is different is that if it happened in testing the first rule set, then the second rule set will not be detected. If one event happened in the first rule set, it can relatively improve performance comparing the double detection.

○ 3 Random testing

Random inspection method is to randomly select one from two rule sets. By this method, it stops the DOS attack caused undefined independent conflict, but that does not guarantee all the events.

In three different detection methods, first hit detection is a kind of intermediate scheme. It helps to prevent DoS attacks by double detection method, and can guarantee produce event of each undefined independent conflict (if possible). However, it cannot guarantee preventing DoS attack as random method could and guarantee to find all events like double check method.

(b)ICMP

ICMP rules are to optimize ICMP rule sets by ICMP type domain. No type domain rules are considered to be general ICMP rules and be added to the rule set. So, when some kind of packet of ICMP type arrives, the corresponding types of rules are applied to proceed. If it has no type field, general rules are applied. For example, ICMP packet of type code 8 (echo request) takes the type code 8 as rule set of parameter selection type 8 and detect it, and other rules don't need to be tested.

(c)IP

Rules optimizer use IP transmission protocol field as independent parameters to optimize the IP protocol [36]. All IP rules in TCP/UDP, or ICMP are broken down into special agreement rule sets. And other transfer protocol domain except for TCP/UDP or ICMP is decomposed into IP rule sets. And rules not including IP protocol domain rules are considered to be general IP rules and are added to TCP/UDP, IP and ICMP rule sets as is shown in Figure 3.

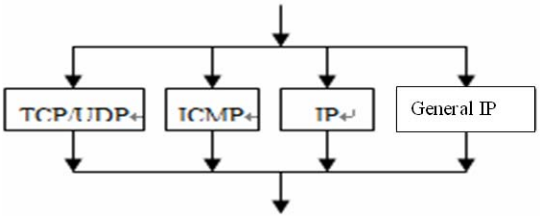

Figure 3. Rules optimization design

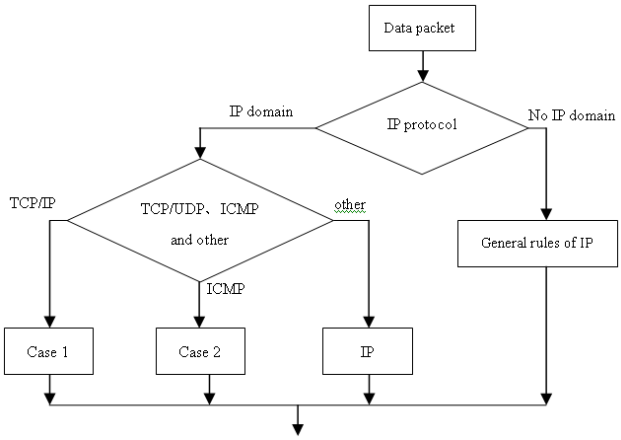

Figure 4. General structure of rule set optimization

Figure 4: General structure of rule set optimization

As is shown in Figure 4, the overall structure optimization of a rule set is constructed, which is more in detail than that in Figure 3. When a packet is got, first determine whether it has IP protocol field, if not, use general IP rules to deal with it; If there is the IP protocol field, then further judge whether it is TCP/UDP, ICMP and other, if it is TCP/UDP, use the corresponding rule set based on independent parameters to dealt with it. If it is ICMP type based on its independent parameters, the corresponding rule set is used, otherwise deal with it by IP rules. According to

the test results, judge whether it is the malicious packet or the occurrence of invasion happened.

III. SIMULATION EXPERIMENT ANALYSIS

Experiment is conducted through our establishment of the C++ program. Evaluation of the model is conducted by using IPv6 network intrusion detection rate, false positive rate and residual rate as performance indexes. Simulation experiment data come from KDD Cup 1999 data set, which collects 7 million network connection records, covering a variety of intrusion data types and normal data. Simulation experiment select data from training data and 10% subdata set of testing data, of which 2000 data strips forming the training set and test set of 3000 data strips, attack types included in training set are less than that of the test set.

A. Data preprocessing

The original data of KDD CUP 99 data set contains two attribute types: character attributes, such as udp and tcp, etc.; symbol type data establishes mapping code to convert; Numerical attributes, such as 105,0,4,0.01. Discrete data performs box dividing process according to data characteristics, which divide data into different boxes, taking the median value of the same box data as their value and data in different boxes has no intersection. Continuous data are processed respectively according to the characteristics of different attributes, and the data process is of standardization.

1) Decimal scale standardization is used in small numerical value data, which is standardized by moving the attribute value of the decimal point position:

Value '= value * 10 j (3)

Such as 0.01 can be standardized to 1, and make it present discreteness.

2) Dividing standardized data into boxes to achieve the effect of data compression.

B. Attribute reduction - dimension reduction treatment

Data of intrusion detection data set comes from network packet information captured, some feature data of which has great contribution in determining if there is an intrusion behavior, and some has no contribution to determine intrusion behavior. These data will make great cost in building concept lattice and the formation of detection rules. Feature attribute dimension reduction treatment must be done.

1) Ergodic data set; select the object of the same attribute values; keep one, and delete all the rest; make the data set without redundant objects.

2) Ergodic data set; delete the column of the same attribute value; reduce attribute dimension.

3) Establish attributes matrix which can be identified by initial form background, and get all reduction from the least disjunctive normal form.

Attribute reduction is conducted in KDD CUP 99 data set. Retain 11 attributes as characteristic attributes constructing detector from 42 feature attributes (41 condition attribute, 1 decision attribute).

C. Results analysis

Every 1000 data strips of test set data are formed a group, and three sets of data are experimented respectively based on the static algorithm (FCAS) based on concept lattice, dynamic optimization algorithm (FCAD) and the method proposed in this paper. The average detection effect of three groups' data is shown in Table 1.

TABLE I. ALGORITHMS EFFECT COMPARISON

Average false negative（%）	Average measurement rate（%）	Average missing report rate（%）
FCAS	12.3	82.5	17.5
FCAD	9.4	89.2	10.8
This paper's method	6.1	93.7	6.3

From the experiment result, the static methods can't timely adjust IPv6 intrusion detector according to the current detection circumstance. The detection rate of dynamic optimization detection algorithm in abnormal behavior tests is obviously higher than that of the static methods and false detection rate lower than that of static detection algorithm. Because two stage detection mechanisms are set up, the attack form which is not detected by the first detector based on concept lattice and artificial immune intrusion detection algorithm may be detected in the secondary test. This technology fusion is to make up for the deficiency of single detection. The detection rate of this paper's method is higher, false negative rate and missing report rate are obviously lower than single detection method, which can meet the requirements of the intrusion detection more.

IV. CONCLUSION

With the rapid development of Internet technology, network structure is becoming more and more complicated; network security has become increasingly important and complicated. This paper puts forward an intrusion detection system under IPv6 platform based on intrusion detection feature attribute reduction by using pattern matching, so as to expand the range of application and user group of the security products. By the analysis and comparison of various pattern matching algorithms, the new algorithm realizes the intrusion feature module matching under IPv6, and make detection system be of high efficiency. Later experiments have proved this view.

Acknowledgment

This research was supported by the National Science Foundation of Jiamusi University(Grant No. L2011-025）, the National Science Foundation of Jiamusi

University(Grant No. L2012-074）, the National Science Foundation of Jiamusi City(Grant No. 12004）all support is gratefully acknowledged.

REFERENCES

[1] Ivan Goethals, Kristiaan Pelckmans, Johan A.K. Suykens, Bart De Moor. Identification of MIMO Hammerstein models using least squares support vector machines[J]. Automatica, 2005, 41(7): 1263-1272.

[2] [8] Everthon Silva Fonseca, Rodrigo Capobianco Guido, Paulo Rogério Scalassara, Carlos Dias Maciel, José Carlos Pereira.Wavelet time-frequency analysis and least squares support vector machines for the identification of voice disorders[J]. Computers in Biology and Medicine, 2007, 37(4): 571-578.

[3] Mohammad Saniee Abadeh, Jafar Habibi, Zeynab Barzegar, Muna Sergi. A parallel genetic local search algorithm for intrusion detection in computer networks[J]. Engineering Applications of Artificial Intelligence, 2007, 20(8): 1058-1069.

[4] Roberto Perdisci, Giorgio Giacinto, Fabio Roli. Alarm clustering for intrusion detection systems in computer networks[J].Engineering Applications of Artificial Intelligence, 2006,19(4): 429-438.

[5] apnik VN.The Nature of Statistical LearningTheory[M].New York:Spring-Verlag,1995.