Research on packet filter rules of the firewall based on Visual Prolog

Wang Yu-gang Ge Yin-mao Yang Jian-xin
Department of aviation armament fire control system, Naval Aeronautical Engineering Academy of Qingdao Branch
Qingdao, 266041, China
flyingmanw@163.com

***Abstract*—The packet filter rules of firewall are established according to the needs of network security, while to manage filter rules becomes more and more complicated, and easy to make mistakes, especially in enterprise network. In order to implement correct policies, the firewall filter rules should be checked and organized carefully. This article studied the relations between firewall filter rules, defined each kind of unusual situation, and through the expert system language Visual Prolog realized the function: to inspect the accuracy of the filter rules to deal with filter rules、 to inspect redundancy and so on. It has made positive effect on enhancing the intelligence of the firewall.(Abstract)**

Keywords-firewallt;filter rules; relevance; redundancy; contradictory rules

I. INTRODUCTION

Firewall as the most commonly used security measures currently, played an important role in practical application. Especially packet filtering firewall, easy to use, the high degree of transparency to user, forward faster, more efficient, which made it applied widespread. However, the traditional packet filtering firewall in dealing with modern network attacks, there exist many shortcomings, especially filtering rules settings, which needs expert-level comprehensive in-depth knowledge In reality, security experts extremely lack. which cause the common network management personnel in firewall settings, because of lack of experience or knowledge, can not be effectively developed network security strategy, leading to many hidden dangers of network security; On the other hand makes packet filtering firewall a lot of errors in use. Moreover, for an organization, only one packet filtering firewall is not enough, but also it should be allocated. Firewall configuration usually use vulgar language to write. How to use the words that a firewall can understand to express the organization security policy, achieving packet filtering firewall filtering rules for the network data flow filter, thus preventing illegal access, At the same time allow legitimate visit successful, and to network attacks, timely and timely manner to interact with the network administrator, to achieve better management of the firewall.

To this end to try to analyze filter firewall rules, through Visual Prolog language to achieve the goal: detection of the unusual rules, to realize the integrity of the firewall rules.

II. TO ANALYZE THE RELEVANCE OF FIREWALL FILTER RULES

A. The definition of the relationship for firewall filter rules

In order to analyze the firewall rules, first analyzing all possible links between filter rules. The filter rules established is based on IP packet, which contains five basic elements: the protocol source address destination address source port and destination port and so on. The relationship between firewall rules are based on the comparison of the five elements. So first of all we learn the definition of the relationship between the rules.

B. Definition of the relationship between the rules

Definition 1: If any domain of the rule R_x is not the subset of the domain, or superset, or the same of the corresponding rule R_y .Then we call R_x and R_y are completely unrelated. That is:

$$\forall i \quad : R_x[i] \triangleright / \triangleleft R_y[i]$$

among: $\triangleright \ \triangleleft \in \{\subset, \supset, =\}$,
i$\in$ {Proto, Src, Dst, Sport, Dport}

Definition 2: If any domain of the rule R_x are equivalent to the corresponding domain of rule R_y . Then we call R_x and R_y are equivalent That is:

$$\forall \mathrm{i}: \ R_x[i] = R_y[i]$$

Definition 3: If any domain of the rule R_x is the subset of the domain, or the same of the corresponding rule R_y . Then we call R_x and R_y are including relevant. That is:

$$\forall \mathrm{i}: R_x[i] \subseteq R_y[i] \ \text{ and } \ \exists \mathrm{j} \ R_x[j] \neq R_y[j]$$

Among : i, j $\in$ {Proto, Src, Dst, Sport, Dport}

Definition 4: If at least one domain of the rule R_x is the subset of the domain, or superset, or the same of the corresponding rule R_y . Moreover at least one domain of the rule R_x is not the subset of the domain, or the same of the corresponding rule R_y . Then we call R_x and R_y are part of irrespective. That is:

$$\text{i, j: } R_x[i] \triangleright \triangleleft R_y[i] \text{ and } R_x[j] \triangleright / \triangleleft R_y[j] \text{ and } \mathrm{i} \neq \mathrm{j}$$

978-0-7695-3336-0/08 $25.00 © 2008 IEEE
DOI 10.1109/CSSE.2008.1453

Definition 5: If some domain of the rule R_x is the subset of the domain, or the same of the corresponding rule R_y Moreover the other domain of the rule R_x is superset of the domain of the rule R_y. Then we call R_x and R_y are cross-related. That is:

$$\forall i : R_x[i] \triangleright\triangleleft R_y[i] \text{ and}$$

$$i, j \; R_x[i] \subset R_y[i] \text{and } R_x[j] \supset R_y[j] \text{ and } i \neq j$$

C. *Describing examples*

For the definition of firewall filter rules above, examples as follows: It describes all possible relationships of the rules.

Table1: The example of firewall rules:

FN	Action	Proto	Src(src_ip)	Ds（dst_ip）	Sport	Dport
1	Deny	Tcp	140.192.37.20	Any	Any	80
2	Accept	Tcp	140.192.37.*	Any	Any	80
3	Accept	Tcp	Any	162.120.33.40	Any	80
4	Deny	Tcp	140.192.37.20	162.120.33.40	Any	80
5	Deny	Tcp	140.192.37.30	Any	Any	21
6	Accept	Tcp	140.192.37.*	Any	Any	21
7	Accept	Tcp	140.192.37.*	162.120.33.40	Any	21
8	Deny	Tcp	Any	Any	Any	Any
9	Accept	Udp	140.192.37.*	162.120.33.40	Any	53
10	Accept	Udp	Any	162.120.33.40	Any	53
11	Deny	Udp	Any	Any	Any	Any
12	Deny	Tcp	140.192.37.20	Any	Any	80
13	Accept	Tcp	140.192.37.*	Any	Any	80

According to previous definition of relevance, It can be seen from the table,such relation exist in these filter rules. Rule 1 and rule 2 are including relevant, rule 1 is included in rule 2, it is a subset of rule 2, rule 2 is superset. Rule 2 and rule 6 are part of irrespective; Rule 1 and rule 3 is correlative; Thus we can draw two conclusions:

Conclusion 1 All these relations defined above are exclusive, Any of two rules can not satisfy two definition at the same time.

Conclusion 2 All these relations defined above have comprise all possible relations which exist between any filter rules.

III. TO ANALYZE THE RELEVANCE OF FIREWALL RULES

For the rules between firewalls can be summarized into two categories: relevant rules and unrelated rules. Among relevant rules contains: R_x and R_y are equivalent, including relevant, and cross-related three kinds of relationships, And unrelated rules contains: R_x and R_y are completely irrelevant, and part of irrespective two kinds of relationships. Among them, The order of the firewall filter rules relevant is of importance, different order decided a different security strategy, because the process of packet filtering is an order of match which data packet compares filtering rules. Until the match to a rule succeed. If filter rules are completely irrelevant, then the order does not matter. However, usually most of filtering rules are relevant. In such conditions, if the order of relevant rules does not consider carefully, some rules may be shielded by other rules. Especially when many filtering rules exist in an security policy. Conflicting rules or redundant rules relatively raise more. Thus below we classified these abnormal firewall rules defined above.

Below the definition of abnormal firewall rules, include an explicit conflict of error (causing some filter rules have always been shielded), or warning of potential conflicts (implicit in filter rules). Thus can be divided into redundant rules, conflicting rules (here the definition of redundancy and contradictions are according to the action which deals with filter rules). Below we analyze redundant and contradictory rules.

A. *Redundant rules*

1) Repetitive redundancy rules

If two filter rules, each corresponding item of the rule are equivalent (including the action to deal with this filter rule), then we call these two filter rules are repetitive. For repetitive redundancy rules, such as rule 12 is a redundancy rule of rule 1 in Table 1, in this condition we only need to delete redundant rules.

2) Including redundancy rules

If two filter rules implement the same action for the same packets, and to remove any of the rules do not change security strategy, then we call these two filter rules as redundant. That is: if R_x is in front of R_y, and R_y is a subset of R_x or completely equivalent. and also they have the same action to deal with filter rules. Then R_x is redundant of R_y. Such as rule 7 is redundant of rule 6, rule 9 is redundant of rue 10 in Table 1. So in this condition to delete rule 7 and rule 9, does not affect security strategy. Redundancy is considered to be a mistake, and a redundant rule does not work, but it increases the length of filtering rules table, and will increase the time and space to find. So as soon as we find redundant filter rules, we should notify administrator to amend the rules or delete redundant rules.

B. *Conflicting rules*

1) Repetitive contradictory rules

If each of corresponding item in two filter rules is equivalent (including the action to deal with the filter rules).Then we call these two filter rules are repetitive contradictory rules. Such as rule 13 and rule 2 in table 1. For this kind of rules, through interaction with administrator, to determine how to deal with the filter rules and remove the corresponding contradictory rules.

2) Shielding contradictory rules

For two filter rules, If one filter rule matches the other filter rule' all data packets behind. Then the other filter rule was shielded by the front filter rule.

That is: if R_y is behind of R_x, R_y is a subset of R_x, and the two filter rules have the same action how to deal with. Then R_y is shielded by R_x. Such as rule 4 is shielded by rule 3 in Table 1, and rule 4 can not be activated. Shielding is a serious mistake, because the rules shielded does not work, this will cause a mistake that a data packet originally be allowed but was prohibited Therefore while finding shielding we should notify the administrator as soon as possible, and to delete the rules to rectify the error .

3) Concluding contradictory rules

If a rule matches all data packets that another filter matches in the front, then we call these two filter rules are concluding contradictions rules. That is, if R_y is behind of R_x, R_y is superset of R_x, and also they have the different action to deal with the filter rules, then we call R_x and R_y are concluding contradictions rules, such as rule 1 and rule 2 in Table 1. If these two rules change their order , the security strategy has also changed accordingly, and rule 1 can not be activated. Therefore, for this condition we should notify administer.

4) Cross-related contradictory rules

If two relevant filter rules, the first rule matches part of packet that the second filter also matches. And also reverse. Then we call these two filter rules are cross-related contradictory rules. That is if R_x and R_y are related, and they have different action to deal with the filter rules, then R_x and R_y are cross-related contradictory rules. Such rule 1 and rule 3 in Table 1 .If these two rules change their order, there will be a different security strategy. Cross-related contradictory rules is considered to be an important warning, because this indicates a fuzzy security strategy. For example: rule 1 and rule 3 indicate the HTTP traffic which from address 140.192.37.20 to address 161.120.33.4 was denied. However, if these two rules change their order, the same traffic will be adopted. So solving this conflict, according to security strategy, administrators should make their choice.

IV. HANDLING FIREWALL FILTER RULES WITH VISUAL PROLOG LANGUAGE

Below we will use Visual Prolog language to handle with filter rules, thereby find abnormal rules、conflicting rules and potential problems automatically.

Before learning Visual Prolog language first we introduce how to express filter rules. filter_rule contains the predicate: 'mode' and 'packet'.

```
filter_rule(Mode,Packet):-
        mode(Num,Action),
        packet(Proto,Src,Dst,Srcport,Dstport).
```

Among each domain of filter rules as follows:

Num	the order of filtering rules in the list
Action	how to deal with this filter rule (allow / deny)
Proto	protocol (tcp / udp / icmp)
Src	the source ip address of packet
Dst	the purpose ip address of packet
Srcport	the source port of packet
Dstport	the purpose port of packet

For example:

```
filter_rule (mode (1,"deny"), packet ("tcp",
"140.192.37.20","12.12.12.1", "any", "80")).
```

This filter rule means that: the number of filter rule is '1', to 'deny' protocol 'tcp' that from source address '140 .192.37.20' to purpose address '12.12.12.1',and from any source port to purpose port '80'.

A. Handling rules irrelevant with Visual Prolog language

For all firewall rules irrelevant, because their position is not important in the list. That means they do not affect security strategy, so we extract all irrelevant filter rules and put them in front of relevant rules .

To deal with irrelevant rules, We use 'attract' and 'attract all' to attract them and save them in a .txt file by 'save'. At last we put all these irrelevant rules in front of the list.

B. Handling rules relevant with Visual Prolog language

To determine rule R_x and R_y , if R_x is in front of R_y .First we match each corresponding domain, from protocol、source address to purpose address, from source port to purpose port.If any of corresponding domain of filter rules are equivalent, and also have the same action, then R_x and R_y are repetitive redundancy rules. But if they have different action to deal with the filter rules, then R_x and R_y are repetitive contradictory rules. Here we use the clause deal_unique_rel_frules (md, pack) to deal with repetitive redundancy rules.

```
deal_unique_rel_frules(mode(A1,B1), packet(C,D,E,F,G)) :-
        filter_rule(mode(A2,B1),packet(C,D,E,F,G)),
        A1<>A2,
        write("it is filter rule Number:"),
        write("",A1),nl,
        write("itis filter rule Number:"),write("",A2),nl,
```

```
write("UNIQUE_REL(B1=B1) filter rule(for the first)"),nl,

        retractall(filter_rule(mode(A1,B1),packet(C,D,E,F,G)),
                                fred),
 nl.
```

While handling repetitive contradictory rules can use the clause:

deal_unique_rel_frules (md, pack)

If any corresponding domain of R_y is a subset or equivalent of R_x, and also they have the same action to deal with the filter rules. Then R_x and R_y are including redundancy rules. But if they have the different action, then we call R_x and R_y are shielding contradictory rules. If any corresponding domain of R_y is superset of R_x, also they have different action, then R_x and R_y are concluding contradictory rules. To deal with them can use this clause:

deal_include_rel_frules (md, pack)

If any corresponding domain of R_y is a superset or equivalent of R_x, and also reverse. Then R_x and R_y are cross-related contradictory rules.

To deal with them can use this clause:

deal_cross_rel_frules (md, pack)

In different circumstances to operate respectively. If all the above are not met, then there is no abnormal. The basic idea to check abnormal is to check the rules one by one in the list. Then determine which kind of abnormal it belongs to, then to deal with it.

V. CONCLUSION

This paper aims at packet filtering rules in actual allocation of network strategy, On foundation of deep study of the link between filter rules, defined all possible relations and classified them in different kinds. to express and to find exception between filter rules with Visual Prolog language. thereby determine their order in list. This help administrator configure security policy. and provide great convenience, when the administrator insert, delete or modify rules, he can use this method to find correct location automatically and prompt managers to amend the existing exception and modify security policies before and after the change. So it can help administer to eliminate potential safety and other problems because of wrong configuration, so as to enhance the intelligence of the firewall.

REFERENCES

[1] Joshua D.Guttman. Filtering postures: Local enforcement for global policies. In Proceedings of the 1997 IEEE Symposium on Security and Privacy.

[2] An expert system for analyzing firewall rules Pasi Eronen and Jukka Zitting Helsinki University of Technology {pasi.eronen,jukka.zitting}@hut.fi

[3] Wu Kexi, Zhao Qinyan intelligent features of the firewall small micro-computer system in June 1999

[4] Zheng Jinyuan, Han Lingling, Li Xiang object-oriented technology and application of Visual Prolog 6 computer and information technology in 2005 13 vol 5

[5] Lei Yingjie, Hua Jixue the theory of Visual Prolog in deal with backdate computer engineering in 2005 18

[6]Lei Yingjie Visual Prolog programming, environment and interface, Beijing: National Defense Industry Press, 2004

[7] Gao Feng Xu Nanshan the research of packet filtering firewall rules. Computer applications, 2003,23 (6)