Dynamic Rule based Interfirewall Optimization using Redundancy Removal Algorithm

Arun Prasath. Y
M.E (Computer & Communication)
Sri Venkateswara College of Engineering,
Affiliated to Anna University,
Tamilnadu, India

Revathi. N
Assistant Professor
Sri Venkateswara College of Engineering,
Affiliated to Anna university,
Tamilnadu, India.

ABSTRACT

Firewall is a typical security system that extensively secures the private networks. The operation of a firewall is to analyze every packet and decide whether to accept or discard it based upon the firewall policy. This policy is specified as a set of rules. The work focuses on inter-firewall optimization over distinct administrative domain without exploiting the privacy policies. With the massive growth of Internet-based applications, the number of rules in firewalls has been increasing in a rapid rate, which degrades the network performance and throughput. To mitigate the number of rules validation for every session, a dynamic rules estimation algorithm is proposed. However, an error in a firewall either discloses secret information from its network or interrupts proper communication between its network and the Internet. The redundancy removal algorithm is used to overcome these problems by reducing the redundant rules in the firewall with multi-rule coverage. The optimization process involves semi-honest computation between the two firewalls by preserving privacy of the each party firewall policies. The algorithm used will avoid the rules overhead and increases the efficiency by optimizing the firewall.

General Terms

Interfirewall optimization

Keywords

Dynamic rule estimation, Redundancy removal algorithm, Firewall optimization.

1. INTRODUCTION

A firewall is a network security system that filtersthe incoming and outgoing network traffic by verifying the data packets and decide either they should be allowed or discarded, based upon the rule set. A firewall operates as a barrier between a secure and trusted internal network and other networks (e.g., the Internet) that is assumed to be unsecured. Several personal computers include software-based firewalls in order to protect over threats from the public Internet which is of enormous use. Many routers that pass data between networks contain firewall and subsequently many firewalls can perform basic routing functions which are related to that particular network and its function. In terms of computer security, a firewall is a piece of software which monitors the traffic over the network. The rules decide if a packet can be accepted or if it is discarded. In general, a firewall is placed between two or more networks in which one is secured and the other are unsecure. When a large network needs to be protected, the firewall software is implemented on a dedicated hardware, which does nothing else.

A firewall is often placed at the entrance between a private network and the external network so that it can check each incoming or outgoing packet and decides whether to allow or discard the packet based upon its policy. A firewall policy is generally specified as a sequence of rules, called Access Control List (ACL), and each rule has a predicate over multiple packet header fields (i.e., source IP, destination IP, source port, destination port, and protocol type) and a decision (i.e., accept and discard) for the packets that match the predicate. Intrafirewall optimization means optimizing a single specific firewall. It is achieved by means of removing redundant rules, rewriting rules and so on. While interfirewall optimization requires two firewall policies without any protection to its privacy, and thus can only be used within a single administrative domain. Upon receiving a packet, a firewall checks the packet's header against a set of user-specified rules (analyzing) and forwards/drops the packet if it is desired/undesired (filtering).

Through packet analyzing and filtering, firewalls can detect suspicious packets and prevent them from enteringthe network. A firewall can enforce a complete network access if all incoming and outgoing packets are allowed to pass through the firewall which is organized based on its rules and policies. Although packet monitoring and filtering help improve network security, it is important to ensure that they do not disrupt the availability and utility of the overall system. A firewall cannot forward a packet until analysis has finished. Therefore, it will incur additional latency to packets. With limited buffer size, continuous packet analyzing time may also cause the firewall to drop packets unconditionally. The performance of a firewall should not be mitigated when under attack; otherwise its purpose would have been defeated. Although one could expect the rapid advancement of hardware to help alleviate this challenge, hardware upgrades many ways be practical.

In this paper, we will be discussing about the so far works that had been carried out under firewall optimization in section 2. And section 3 is about the proposed dynamic rule estimation algorithm that is developed to generate temporary, equivalent and compact firewall. Section 4 consists of proposed redundancy removal algorithm to discard redundant rules during rules verification which is carried out by the firewall. While the operation that is performed by the proposed algorithms are explained in section 5. The result comparison and analysis done so far is illustrated in section 6. Two algorithms are designed that allows two adjacent firewalls to achieve inter-firewall optimization with respect to each other without knowing the policy of the other firewall.

2. BACKGROUND

Many companies and organizations use firewalls to segment access control within their own networks. Firewalls are typically deployed to filter traffic between trusted and untrusted zones of corporate networks, as well as to police their incoming and outgoing interaction with other networks – e.g., the Internet. To solve conflicts when processing packages, most firewall implementations use a first match strategy through the order of rule. Hence, every packet filtered by the firewall is mapped to the decision of the best priority rule. In fact, many of these policies, rules and objects in a standard firewall or router policy are outdated. These outdated rules represent a potential security risk and has to be eliminated, which is nearly impossible for managers to detect them and remove them without risking business continuity.

To begin with, the prior work was done at 2004 by [1] which focuses on rule sets for Check Point's Firewall 1 product and specifically on 12 possible misconfigurations that would allow access beyond a typical corporation's network security policy. However, this ideology is static and applicable only over certain aspects. Later in 2006, [2] dealt with optimize packet classifier configurations by identifying semantically equivalent rule sets that lead to reduced number of TCAM entries when represented in hardware. Here the packets were classified using a specific packet classifier, TCAM which ensures the semantic of the firewall rules. In 2007, [4] proposes a systematic approach, the TCAM Razor, that is effective, efficient, and practical than that of TCAM regarding range specification issue. This technique introduces an upgraded TCAM with more efficiency when compared to [2] work. In 2008 [6] proposes the interval expansion problem of TCAMs can be addressed by individual packets specifications in packet classifiers. This equivalent transformation can significantly reduce the number of TCAM entries needed by a packet classifier. Besides, the decision made by the TCAM entry, the number of entries that are entered is considerably reduced.

In 2008, [7] proposes framework that can significantly reduce the number of rules in a firewall while keeping the semantics of the firewall unchanged. This technique considerably increased the throughput when compared to its previous works. At the end of 2008, [5] proposes the method of diverse firewall design, which is inspired by the well-known method of design diversity for building fault-tolerant software. Here the design of firewall was done in a diverse manner such that firewall operates depending upon the type of service is used. Later in 2009, [8] proposed bit weaving is based on the observation that TCAM entries that have the same decision and whose predicates differ by only one bit can be merged into one entry by replacing the bit in question with *. This method detects the semantically equivalent rules and replaces them as a single rule, by entering * in the other rules decision. In 2010, [10] proposed the re-encode of the entire classifier by considering the classifier's decisions rather than re-encode only ranges in the classifier ignoring the classifier's decisions as prior work does, which completely depends upon the encoding done to the packet classifier. Later in 2013, [13] proposed a mathematical way of estimating the firewall rules to improve the inter firewall optimization, involves the exponential estimation of the semantically equivalent rules, that can be discarded by the firewall. Prior work over firewall optimization focuses on either interfirewall or Intrafirewall optimization within a single administrative domain where the privacy of firewall policies is not a concerned aspect.

3. DYNAMIC RULE ESTIMATION ALGORITHM

The firewall operation is modified by introducing this dynamic rule estimation algorithm. The algorithm works on the basis of estimating the very first packet of the session in such a way that, depending upon the source ip address, source port, destination ip address, destination ip and decision, the algorithm will generate an equivalent but more compact firewall such that, it will reduce the total number of rules that are to be verified. The algorithm can be formulated as follows,

ALGORITHM: Dynamic Rule Estimation Algorithm

INPUT: A firewall ($r_1 \ldots r_n$)

OUTPUT: An equivalent but more compact firewall (FW2)

```
STEP 1: for i = 1 to n do
            equiv[i] := false.
STEP 2: for i = 1 to n do
            if there exist rule $r_k$ in the firewall, where $i < k < n$,
        such that the following three conditions hold,
                        (a) equiv[i] = false.
                        (b) $r_i$ and $r_k$ have the same decision.
                        (c) packet satisfies both $r_i$ and $r_k$.
                                then equiv[i] := true.
            else equiv[i] := false.
STEP 3: for i = 1 to n do
            if equiv[i] = true.
            then remove $r_i$ from the firewall
```

The algorithm estimates the entire available rule sets for the first incoming packet so that the verification of the rules for the forthcoming packets will be done over the extracted rules, which are related to them. The verification of unwanted rules is avoided and through this way, the optimization between two distinct domains is achieved without disclosing the firewall policies between each other. The algorithm generates an equivalent but more compact firewall which is a volatile for every session. In this paper, we consider FW2 as the temporary, equivalent but more compact firewall. Once the session gets finished or the request comes from some other client, the compact firewall will be discarded and a new temporary, equivalent and compact firewall is generated by the algorithm based upon the incoming packets.

Managed policy is difficult to maintain and requires the attention of senior administrators with need of expert, undocumented knowledge. Since a mistake can result in application or network downtime, which is unfeasible to assign policy management to less-experienced or outsourced staff. The probability of an error and cost is even higher for security service providers. In addition to security risks, a weaker firewall policy can have a major impact over performance. The complete rule base will be parsed from top to bottom and it grows, as hardware requirements increase. Security teams require automation in order to maintain efficient and secure policies on all of their firewalls and routers. A firewall protects one part of the network against invalid access. Optimizing firewall operations is necessary for improving network performance throughput. Especially, for the two neighbouring firewalls belonging to two distinct administrative domains, the algorithm can identify in each firewall the rules that can be removed because of another firewall. While intra-firewall redundancy removal is already complex, inter-firewall redundancy removal with the privacy-preserving requirement is even harder.

4. REDUNDANCY REMOVAL ALGORITHM

Firewalls have been commonly implemented over the Internet for securing individual networks. A firewall checks each and every outgoing and incoming packet to decide whether either to allow or discard them based upon its policy. Optimizing firewall policies is essential for improving network performance. Here, FW2 is the temporary, equivalent and compact firewall, generated by Dynamic rule estimation algorithm. It explores interfirewall optimization across administrative domains for the first and foremost time. The crucial challenge is that firewall policies cannot be disclosed over the different domains because a firewall policy contains private information and even crucial security holes, which can pave way to the attackers to launch precise attacks. In firewall, the similarity join consists of grouping pairs of records whose similarities greater than a threshold, Privacy preserving algorithms for similarity join are used to protect the data of two sources from being totally disclosed during the similarity join process.

ALGORITHM: Redundancy Removal Algorithm

INPUT: Packets that satisfies a specific rule from FW1

OUTPUT: Allow or discard packets based upon specific rule at FW2

```
1Initialize;
2 while the incoming packet from FW1 do
3       for each rule $r_i$ in FW2 do
4 compare if $r_i$ gets satisfied with any rule$r_k$ FW2
5       for each packet do
6 decide upon every packet that satisfies $r_k$
7 else do
8$r_k$ = Ø;
9Obtain new FW2 for the incoming packet and go to step 2
10return;
```

FW1 - Firewall of network 1 (packet outgoing interface)
FW2 - Firewall of network 2 (packet incoming interface)

In the simplest case, when the joint operation is done on two sources, A and B, source A is not supposed to know the content of all the records in source B. Instead, source A can know the content of the records that will be joined with its own records. The records in source B that will not be joined with source A will be hidden from it, and vice versa. Even though similarity join have a wide range of applications in various fields, only a few researchers have concentrated on performing similarity join under privacy constraints. The key contributions are made such as a novel redundancy removal algorithm for detecting inter-firewall redundant rules in one firewall with respect to another firewall is proposed. It represents the effort along this unexplored direction. The communication cost is fewer than a few hundred KBs. The algorithm requires no additional online packet analyzing overhead and the offline analyzing time is comparatively less than a few hundred seconds to reduce the redundancy and to improve the optimization.

5. INTERFIREWALL OPTIMIZATION

Interfirewall optimization requires two firewall policies without exploiting its privacy and this can only be used within a single administrative domain. But, it is common that two firewalls belong to different administrative domains where firewall policies cannot be shared with others. Keeping firewall policies secret is very important for two reasons. They are discussed as follows,

- First, a firewall policy may have security holes that can be exploited by attackers. Analytical studies have shown that most firewalls are misconfigured and have security holes.
- Second, a firewall policy often contains confidential private information, e.g., the IP addresses of servers, which can be exploited by the attackers to perform more targeted and precise attacks.

To explore the basic problems of interfirewall optimization across different administrative domains depends upon the firewall policies of the specific domain. The key crucial challenge is that firewall policies cannot be shared across domains because a firewall policy contains privacy information and even crucial loop holes, which can be targeted by attackers. The semi-honest model for different administrative domain and privacy-preserving cooperative firewall policy optimization algorithm has to be processed to overcome the problems of firewall. Specifically, focus on removing inter-firewall policy redundancies in a privacy preserving pattern. To determine whether a rule in FW2 is interfirewall redundant with reference to FW1, Network 2 certainly needs some information about FW1; whereas Network 2 cannot reveal FW1 such information Consider two adjacent firewalls FW1 and FW2 belonging to different administrative domains Network 1 and Network 2.Let us assume that, FW1 acts as packet outgoing interface and FW2 acts as packet incoming interface. For a rule r from FW1, if the same rule r is available in FW2 then the packets are analyzed only with that rule and decision is made accordingly. The rule r can be changed if any of the incoming packets violates it. In such situation, it implies that FW1 have changed the rule r and the same rule r should be searched again in FW2. This rule r is referred here as inter-firewall redundant rule. Note that FW1 and FW2 only filter the traffic from FW1 to FW2; the traffic from firewall 2's outgoing interface to firewall 1's incoming interface is guarded by the specific mechanism.

Routing rules establish static routes in the firewall. Firewall software varies widely in the way it can process packets. For example, some firewalls first perform port and address transformations and then apply policy rules, while some others does it the other way around. Design an algorithm that allows two adjacent firewalls to identify the inter-firewall redundancy with respect to each other without knowing the policy of another firewall. While Intrafirewall redundancy removal itself is already complex, interfirewall redundancy removal with the privacy-preserving requirement is even harder. To determine whether a rule in FW2 is interfirewall redundant with reference to FW1, Network 2 certainly needs some information about FW1; whereas Network 2 cannot reveal FW1 such information.

6. RESULTS AND DISCUSSIONS

Firewall basically deals with detecting the each incoming and outgoing packets based on the rules provided by the administrator. The efficiency of firewall can be increased by optimizing it. Firewall optimizations focus on both intra and inter firewall where privacy policy is of concerned. The technical challenge deals with the redundant rules removal when performed over a network. Each physical interface of a firewall is concerned with two ACLs: one for filtering outgoing packets and the other one for filtering incoming packets. The rules in a firewall policy typically follow the first-match strategy where the decision of the rule is the decision for the packet matches over the policy.

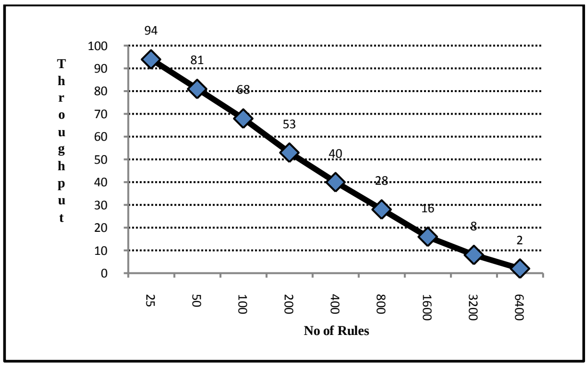

Figure 1: Performance Evaluation

Here we use the terms firewalls, firewall policies, and ACLs interchangeable. The number of rules in a firewall is always inversely proportional to its throughput. The proposed redundancy removal algorithm is found to reduce the overall access time of the firewall rules, by means of the interfirewall optimization, which improves the efficiency of the firewall.

7. CONCLUSION

Firewall basically deals with detecting the each incoming and outgoing packets based on the rules provided by the administrator. The efficiency of firewall can be increased by optimizing it. Firewall optimizations focus on both intra and inter firewall where privacy policy is of concerned. The technical challenge deals with the redundant rules removal when performed over a network. The increase in number of rules in a firewall will automatically decrease the throughput. Thus, our algorithm focuses on reducing the number of rules to be verified by the firewall. Hence, they achieve the interfirewall optimization over two distinct administrative domains without disclosing the firewall policies. The dynamic rule estimation algorithm and redundancy removal algorithm are developed for rule based interfirewall optimization over one firewall with respect to another firewall. The redundant rules are detected when it is implemented in cross administrative domain by ensuring its privacy policy. The algorithm requires no additional online packet analyzing overhead and the offline analyzing time is lower than a few hundred seconds to reduce the redundancy and to improve the firewall operation.

8. REFERENCES

[1] A. Wool, Jun. 2004, “A quantitative study of firewall configuration errors,” Computer, vol. 37, no. 6, pp. 62–67.

[2] Q. Dong, S. Banerjee, J. Wang, D. Agrawal, and A. Shukla, 2006, “Packet classifiers in ternary CAMs can be smaller,” in Proc. ACM SIGMETRICS, pp. 311–322.

[3] L. Yuan, H. Chen, J. Mai, C.-N. Chuah, Z. Su, and P. Mohapatra, 2006, “Fireman: A toolkit for firewall modelling and analysis,” in Proc. IEEES&P, pp. 199–213.

[4] C. R. Meiners, A. X. Liu, and E. Torng, 2007, “TCAM Razor: A systematic approach towards minimizing packet classifiers in TCAMs,” in Proc.IEEE ICNP.

[5] A. X. Liu and M. G. Gouda, Sep. 2008, “Diverse firewall design,” IEEE Trans.Parallel Distributed Syst., vol. 19, no. 8, pp. 1237–1251.

[6] A. X. Liu, C. R. Meiners, and Y. Zhou, 2008, “All-match based complete redundancy removal for packet classifiers in TCAMs,” in Proc. IEEEINFOCOM, pp. 574–582.

[7] A. X. Liu, E. Torng, and C. Meiners, 2008, “Firewall compressor: An algorithm for minimizing firewall policies,” in Proc. IEEE INFOCOM.

[8] C. R. Meiners, A. X. Liu, and E. Torng, 2009, “Bit weaving: A non-prefix approach to compressing packet classifiers in TCAMs” in Proc. IEEEICNP, pp. 93–102.

[9] A. X. Liu, C. R. Meiners, and E. Torng, Apr. 2010, “TCAM Razor: A systematic approach towards minimizing packet classifiers in TCAMs,”IEEE/ACM Trans. Networks, vol. 18, no. 2, pp. 490–500.

[10] A. X. Liu and M. G. Gouda, Apr. 2010. “Complete redundancy removal for packet classifiers in TCAMs,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 4, pp. 424–437.

[11] Fei Chen, Bruhadeshwar.B, A. X. Liu, June 2013,”Cross domain privacy preserving firewall optimisation” IEEE/ACM Transactions On Networking, Vol. 21, No. 3, pp. 857-868.