ENHANCING NETWORK SECURITY AND PERFORMANCE USING OPTIMIZED ACLS

Kuldeep Tomar[1] and S.S Tyagi[2]

[1]Research Scholar, Department of CSE, MRIU, Faridabad, Haryana, India
[2]Professor & Head, Department of CSE, MRIU, Faridabad, Haryana, India

ABSTRACT

Access Control list plays a very important role in network security. Proper combination of rules for ACLs can close loop holes in the system, this minimizing security breaches. An ACL can improvise network performance up to a good level by limiting the traffic controls the areas that can be accessible to any device or user. However, if ACL is not managed properly and efficiently it causes packet latency and degrades the network performance. In this paper we present various optimization mechanisms to achieve an optimal ACL which reduces the Packet latency. We also proposed an efficient optimization algorithm to optimize the ACL to enhance network performance. We also discuss the importance of ACL and the various rule anomalies.

Keywords

Network, Security, ACL. Anomalies, Network Access Control, Optimization, Router, latency, Rule.

1. INTRODUCTION

In the past decade there has been an enormous growth in the usage of computer for data processing and transfer over the network, not only large organisations but even small scale organisations are using them to a large extend. By this increase the incidents of security breaches [1] threats also has increased. Due the dynamic environment there is a need of proper configuration of security tools, development of advanced algorithms and counter mechanism is required. In networking, Routers play a vital role for preventing internal devices, systems and resources secure. Routers are responsible for forwarding packets from source address to destination address. Another important role of the router is to determine whether the packet is authorized by the network administrator to reach the desired destination, because many of intruders exploit packets for performing security breaches. This can be done by using exploiting improper configured ACL (Access Control List). The ACL is a list of rules to determine whether to forward or discard a packet i.e. it acts as a packet filter. They are configured by network administrator to provide additional security to the internetwork. The ACL can also used for enforcing policies such as NAT (Network address Translation) and Traffic Shaping [2]. Proper ACL configured Router's examines each packet to determine whether to forward or drop the packet, this decision is based on the rules specified within the access lists leading to secure network and packet flow.

2. BACKGROUND AND RELATED WORK

Many Valuable contributions have been made by many researchers on the optimization of packets flow on Networks using ACLs on routers etc. But sometimes the settings of improper ACLs may

DOI:10.5121/ijfcst.2014.4602

create loop holes in the system and creating much vulnerability so still there is a great need of deploying proper configured ACLs policies based on algorithms. We studied many paper in the area but we study and get inspired by the valuable work of the following:

William Mahoney, James Harr in July 2010 [3], describes that in Linux or UNIX systems ACL rules are blended with new format and old format of rules, which sometimes tends to create problems in understanding and appropriate implementation of rules. These improper setting of rules can create loop holes in the system leading to vulnerabilities. They use the simplistic rules or permission methods for ACL check of windows and to apply them in the Linux file system.

Wenjuan Xu, Mohamed Shehab, Gail-Joon Ahn, in their paper [4], "Visualization-based policy analysis for SELinux: framework and user study", proposes a framework for SELinux that provides facility of policy violation identification and visualization policies of security for system administrators in form of visuals. They also implemented tool for analysis of policies and generated results based on experimentation.

A Bobyshev, P DeMar, D Lamore, Fermilab, Batavia, in their paper [5] "Effect of Dynamic ACL (Access Control List) Loading on Performance of CISCO Routers", uses their results of their experiments, for dynamically setting of different types of ACLs to improvise network infrastructure and performance. They discussed and experimented how frequently ACLs should be updated, updates of passive versus active ACL, and how frequently the updates should be downloaded on routers so it must not effect CPU utilization of routers, etc.

Liu Zhian, in his paper [6], "Study of Network Optimization Method Based on ACL", provides his valuable contribution by his experiments and mathematical analysis of transferring packets. He use proper applications and load variance of network leading to optimized smooth flow of network. He studied two methods of application on comparison and ACL designing. His results enhance optimized transfer of packets over network.

3. IMPORTANCE OF ACL

ACL provides a very powerful way to control network traffic into or out of a network. Access Control List is important as it facilitates the network administrator by:

- Providing traffic flow control by restricting unwanted routing updates.
- Controlling the areas accessible to a client by restricting the use of network by certain users or devices.
- Increasing network performance by limiting network traffic.
- Providing additional security by restricting the unauthorized packets.
- Controlling type of traffic by filter packets flow in or out of router interfaces.

In spite of all the above, if the ACL is not created effectively , it may add significantly to packet delay and even small ACLs will contribute to this latency simply by their aggregation across several routers[7, 8]. An Example of ACL is given in Figure 1 below.

```
access-list 16  permit permit ip 191.165.1.0  0.0.0.255

access-list 16 deny tcp any any range 125 135

access-list 16 16 deny udu any any eq 1544

access-list 16 permit tcp any any

access-list 16 permit ip 191.164.16.0.0.0.0.255  any

: :

: :

{access-list 16 deny  all} {implicit}
```

Fig.1. Example of ACL

4. RULE

An ACL is a ordered sequence of rules. Each rule is applied to a packet being processed, router forward or deny the packet based on whether the packet match the rule or not. A rule is shown in Fig. 2 below.

```
access-list 7  permit ip 196.44.168.8  any0.0.0.7  any
```

Fig. 2 Simple Rule

A rule consists of six fields:

(1) Action, which could be either permit or deny.
(2) Protocol, such as IP, TCP, UDP, ICMP etc.
(3) Source addresses range, in the form of an IP address and a wild card mask.
(4) Source port range.
(5) Destination addresses range.
(6) Destination port range.

A typical rule fields defined in access lists are source addresses of the packets, destination addresses of packets, or packets upper layer of protocol. Though specific set of policies or rules are defined for protocols. Whenever a rule is added to a ACL that will be appended to the end of the access list statements. A rule cannot be deleted individually, for that ACL has to be deleted.
Network traffic is compared with the access list in a sequential order until a match is found; if matches found then NO further comparisons are made. There is an implicit “deny all” statement at the end of each access list.

deny all {implicit}

If a packet does not match in the access list, it will be denied by the router due to this reason an ACL must have at least one permit statement all traffic is blocked. The working of Access Control List is explained in Figure 3 below.

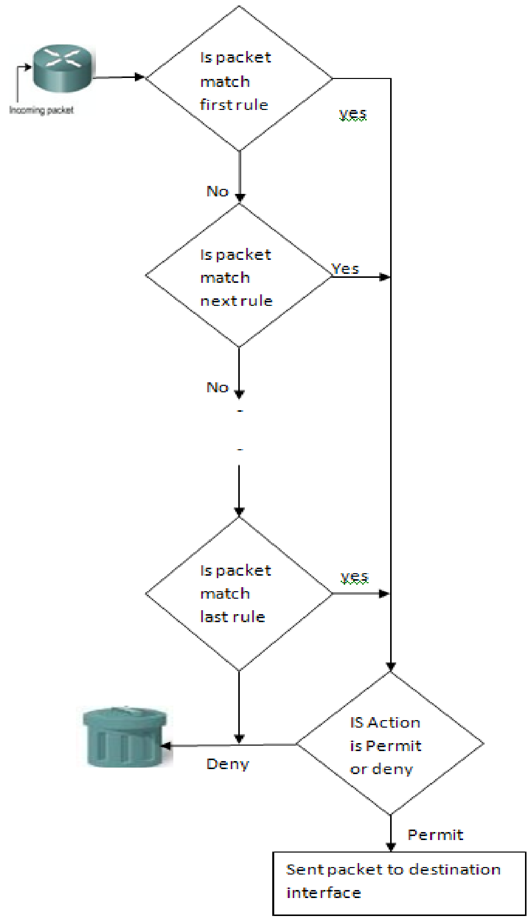

Fig.3. ACL Working

5. RULE ANOMALIES

The large ACL may have many conflicts among rules (i.e. Anomalies) like Redundancy, Inconsistency, Shadowing, generalization etc. There are many possible anomalies or conflicts between almost all rule pair, for example rule correlation, rule shadowing, redundancy, generalization and superimposing [9, 10]. It also affects the network performance. These are explained by M. Al. Abdulmohsin [11]. We are explaining these anomalies by taking simple examples in the following sections. These rule anomalies are explained as follows:

5.1 Redundancy Rule:

The actual meaning of redundancy is the repetition of a rule. Two Rules Rs and Rt are said to be redundant if they match the same network traffic. It include the case, Rs = Rt
Rs: access-list 10 permit ip X, Y eq http;

Rt: access-list 10 permit ip W, Z eq http;
Rs and Rt are redundant, If (s < t) and X is the Superset of W and Y is the Superset of Z.
A redundant rule can be easily moved from the ACL without modifying the semantics of ACL.

5.1.1 Rule Overriding:

Two Rules namely, Rs and Rt where Rs precedes Rt; the Rs Overrides Rt If and only if the type of action is just opposite of each other.
Rs : access-list 10 permit ip X,Y eqssh;
Rt : access-list 10 deny ip X,Y eqssh;
To resolve this conflict the rule which comes after another is to be removed as the ACL rules are processed in sequence from top to bottom.

5.2 Shadowed Rule:

When a Rule permit specific network traffic and a preceding rule deny that network traffic, than the rule is said to be a shadowed rule.
A rule, Rs shadows a rule, Rt where, (s < t) and Rs is the superset of Rt. The Rt (shadowed rule) will never executed as all the fields of Rs are the superset or equal to the respective fields of Rt.

5.3 Covered Rule:

A Rule, Rs is said to be covered by Rule,,Rt if and only if, (s < t) and (Rs.act = Rt.act) and Rt is the superset of Rs.
A covered rule can be easily removed without changing the semantics as a more generalized ACL rule is exist their which still process the same network traffic without harming any security requirements.
There are many algorithms exist that analyzes the conflicts among more the two ACL rules or more, but they do not able to detect all possible conflicts that are present (mainly correlation) [12, 13].

6. THE PROBLEM STATEMENT

Network administrator can easily add new ACL entries (ACE) according to its needs. Due to this the size of ACL grows and when the network traffic comes to router it has to compare every packet with each ACE (or rules) to see if there is a match exist between the packet criteria and ACE criteria. This process is continuing with every rule in sequence until a match is found or the last ACE i.e. Deny ip any anycomes. There is No problem in case of small ACL but what happen if the ACL was around 150 lines or larger then it becomes a problem as the router check each packet with all the rules of ACL [14]. This significantly leads to the increase in the packet delay timings and also increases latency on packet forwarding through the router.

7. OPTIMIZATION TECHNIQUES

In optimization, the main aim is to create an optimized ACL from a already existing ACL. An optimized ACL is a list of rules that fulfils all the security measures and needs a very less computational delay and CPU utilization. The standard or the metric used to measure the optimization is Expected Packet Latency (EPL) [11]. As more reduction in the EPL reflects more better is the optimization techniques. Packet Latency (PL) is the delay in forwarding packet while it is processed by Rule k. The formula for calculating the Expected Packet Latency is given in Figure 4 below:

$$PLE = \sum_{0}^{m-1} Hit(k).PL(k)$$

Where m is the total of ACE's in ACL

Fig.4. Packet Latency Formula

There are two mechanisms for ACL rule's Optimization, as written below:

- ACL optimization: By reducing size of the ACL
- Hits optimization: By Re-ordering the ACL rules based on Hit counts etc.

In the first method, we find conflicts among rules (Redundancy, Inconsistency, shadowing etc) and try to resolve these conflicts either by deleting a rule, merging rules. Due to these actions the size of ACL reduce and as the ACE are less it takes less time in processing a packet and will lead to reduction in packet latency.

In the second method, the rules are arranged in such a manner that the rules which are frequently matched with the packets will be placed on the top of the access list so that it can reduce the packet latency.

7.1 Optimizing ACL: First mechanism

In this method the ACL is minimized by removing the ACL rules or combining some rules by wildcard masking if possible. So this method will decrease the number of Rules that is to be matched with incoming packet. Now the question arises: which Rules are removed or merged? The answer of this is we have to remove rules such as redundant rule, shadowed rule; covered rule etc which if removed doesn't change the semantics of ACL. This is explained as follows:

7.1.1 By Removing Redundant rules:

The redundancy in ACL rules is already described in previous section. Here we elaborate this with an example. Here is an ACL given in Figure 5 below:

```
R1: access-list 10 permit tcp from 172.16.1.1

R2: access-list 10 permit tcp from 172.16.1.1

R3: access-list 10 deny ip from 172.16.1.15

R4: access-list 10 permit ip from 172.16.1.20
```

Fig.5. Original ACL

In this ACL the Rule R1 and R2 are redundant rules. So the list can be optimized by removing redundant rule as shown below:

```
R1: access-list 10 permit tcp from 172.16.1.1
R2: access-list 10 deny ip from 172.16.1.15
R3: access-list 10 permit ip from 172.16.1.20
```

Fig.6. Optimized ACL

7.1.2 By Removing Covered rules:

The covered rule is already described in previous section. Here we elaborate this with example. The ACL example is given below:

```
R1: access-list 10 permittcp from 172.16.1.1
R2: access-list 10 permitip from 172.16.10.5
R3: access-list 10 permitip from 172.16.10.0/0.0.0.255
```

Fig.7. Original ACL

In this ACL, the Rule R2 is covered by Rule R4. So the R2 can be safely removed. The optimized ACL is given below:

```
R1: access-list 10 permit tcp from 172.16.1.1
R2: access-list 10 permit ip from 172.16.10.0/0.0.0.255
```

Fig.8. Optimized ACL

7.1.3 By Removing Shadowed rules:

The shadowed rules are already discussed in previous section. Here is an example in which the ACL is optimized by reducing shadowed rule. The example is given below:

```
R1: access-list 10 permittcp from 172.16.1.1
R2: access-list 10 deny tcp from 172.16.1.1
R3: access-list 10 deny ip from 172.16.1.15
```

Fig.9. Original ACL

In this ACL the Rule R2 is a shadowed rule, it can be safely removed. The optimized ACL is given below:

R1: access-list 10 permit tcp from 172.16.1.1

R2: access-list 10 deny ip from 172.16.1.15

Fig.10. Optimized ACL

By merging mask-able rule address ranges. The ACL given in example below can be optimized by masking the contiguous address ranges by using wildcard mask.

R1: access-list 10 permit ip from host 172.18.10.8

R2: access-list 10 permit ip from host 172.18.10.9

R3: access-list 10 permit ip from host 172.18.10.10

R4: access-list 10 permit ip from host 172.18.10.11

R5: access-list 10 permit ip from host 172.18.10.12

R6: access-list 10 permit ip from host 172.18.10.13

R7: access-list 10 permit ip from host 172.18.10.14

R8: access-list 10 permit ip from host172.18.10.15

Fig.11. Original ACL

In this ACL the address range of all rules is contiguous and mask-able. This can be optimized by merging rules as explained in our propose work.

R1: access-list 10 permit ip from host 172.18.10.8/0.0.0.7

Fig.12. Optimized ACL

8. PROPOSED WORK

Based on the previous sections, we proposed a improved two level optimization technique. We assume that the ACL contains n rules, where R0 is the first rule and Rn-1 is the last rule. In this firstly the ACL is optimized based on the hit counts of the rules of ACL. These Hit Counts are maintained by the network administrator. This technique simply using a sort technique to traverse the ACL and then a rule say Rk is selected for which k is taking values from 0 to n-3 rule and next rule say Rj is selected for which j is taking values from k+1 to n-2 rule. This technique is comparing the hit count of rule Rk with the hit count of rule Rj, if the hit count of Rj is greater than Rk than Rk and Rj are swapped otherwise rule this process of comparison continues till the second last rule.

This reordered ACL is then again optimized by checking for the rule anomalies and removing the rule if they do not change the semantics and security majors. After deleting rules the ACL boundaries are updated. Finally we get the optimized ACL.

In the improvised optimization algorithm, removal of redundant rules, covered rules and update ACL boundaries are not expanded as already explained by M. Al. Abdulmohsin [11] in his paper. The algorithm for optimization is given below:

ALGO: Optimization Algorithm

```
//initialize ACL with Ro as first rule and Rn-1 as last rule.
Step 1:  Repeat
For k: =0 to n-3
Step2:   Repeat
For j: =k+1 to n-2
Step 3:  If hitcnt[Rk] >hitcnt[Rj]
Then
If j <= n-2
            Goto step 2
Step 4:  Else
Swap (Rk,Rj);
Step 5:  If j <=n-2
Then Goto step 2
Step 6:  Else

If k <= n-3
            Then
            Goto step 1
Step 7:  Repeat
For s: = 0 to n-3
Step 8:  Repeat
For t: = k+1 to n-2
Step 9:  If (Rs overrides Rt)
Then
Remove rule Rt and update ACL boundaries
Step 10: If (Rs redundant to Rt)
Then
Remove rule Rt and update ACL boundaries
Step 11: If (Rs covered Rt)
Then
Remove rule Rs and update ACL boundaries
Step 12: If t <= n-2
Goto step 8
Step 13: If s <=n-3
Goto step 7
```

The time complexity for this algorithm is O (n4). This algorithm optimizes the given ACL twice. Firstly by reordering of rules on the basis of their hit counts, Secondly the reordered ACL is reduced in size by detecting and removing rule anomalies. The rules deleted are such that they do not cause any security holes. This leads to the reduction in packet latency as the rules with high hit count are placed on the top of the ACL hence the chances of packet matching with rules are high. Moreover as number of rules are reduced so it takes less time in matching the packet with

ACL entries. Hence increase the network performance and would be able to allow genuine packets in the network and thus leading to enhanced network flow and security.

9. CONCLUSION AND FUTURE WORK

In this paper we present improved mechanisms for optimizing Access control lists. These methods have some pros and cons. We done optimization of ACL manually by using these mechanisms but it can also be done by a router or an application such as ACL Manager. Our proposed optimization algorithm is a combination of both the optimization mechanisms. It would be more beneficial for a commercial network. In our further work we would try to apply more appropriate combinations of algorithms and ACL rules on network edge devices like routers, ASA etc, and also would use resource optimization and network throughput etc of the network leading to more effective packets flow and security.

ACKNOWLEDGEMENTS

The authors would like to thank all the researchers and the authors of referred papers for their contribution in this area.

REFERENCES

[1] Harshita B, N Ramesh, “A Survey of Different Types of Network Security Threats and its Countermeasures”, International Conference on Electrical, Electronics and Computer Engineering, May 2013. Mysore, ISBN: 978-81-927147-3-8.
[2] A. Velte and T. Velte, “Cisco: A Beginner’s Guide”, McGrawHill Inc, 3rd edition (2004).
[3] William Mahoney, James Harr, “A Linux Implementation of Windows ACLs”, International Journal of Computer Science and Network Security, VOL.10 No.7, July 2010.
[4] Wenjuan Xu, Mohamed Shehab, Gail-Joon Ahn, “Visualization-based policy analysis for SELinux: framework and user study”, Int. J. Inf. Secur. (2013). Published online: 8 November 2012, Springer-Verlag Berlin Heidelberg 2012.
[5] A.Bobyshev, P.DeMar, D.Lamore, Fermilab, Batavia, “Effect of Dynamic ACL (Access Control List) Loading on Performance of CISCO Routers”.
[6] Liu Zhian, “Study of Network Optimization Method Based on ACL”, Published by Elsevier Ltd. Selection and/or peer-review under responsibility of [CEIS 2011].
[7] V.Grout, J. McGinn, and J.Davies, “Real-Time Optimization of Access Control Lists for Efficient Internet Packet Filtering”, Journal of Heuristics, Vol. 12, 2006.
[8] V. Grout, J. McGinn, “Optimisation of Policy-Based Internet Routing using Access-Control Lists”, Centre for Applied Internet Research (CAIR), University of Wales.
[9] E.S. Al-Shaer and H.H. Hamed. “Firewall policy advisor for anomaly discovery and rule editing” In Integrated Network Management, 2003. IFIP/IEEE Eighth International Symposium on, pages 17-30, march 2003.
[10] Zhe Chen, ShizeGuo, and Rong Duan. Research on the anomaly discovering algorithm of the packet filtering rule sets. In Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on, pages 362-366, sept. 2010.
[11] Ibrahim M. Al Abdulmohsin, “Techniques and Algorithms for Access Control List Optimization Efficient Internet Packet Filtering”. Computers & Electrical Engineering 01/2009; 35:556-566.
[12] S. Pozo, A.J.Varela-Vaca, and R.M. Gasca. A quadratic, complete, and minimal consistency diagnosis process for firewall ACL’s. In Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on, pages 1037-1046, April 2010.
[13] S. Pozo, R. Ceballos, R. M. Gasca, A. J. Varela Vaca, “Fast Algorithms for Local Inconsistency Detection in Firewall ACL Updates”, DOI: 10.1109/SECURWARE. 2008.40 In proceeding of: Emerging Security Information, Systems and Technologies, 2008. SECURWARE '08. Second International Conference on Source: DBLP, Publisher: IEEE.

[14] Large Example of a Border Gateway Router ACL [Online]: http://www.rpatrick.com/tech/acl/
[15] J. B. Shukla, G. Singh, P. Shukla, and A. Tripathi, "Modelling and analysis of the effects of antivirus software on an infected computer network," Applied Mathematics and Computation, vol. 227, pp. 11-18, 1/15/ 2014.
[16] Monire Norouzi, Saeed Parsa, Ali Mahjur, “A New Approach For Formal Behavioral Modelling Of Protection Services in Antivirus Systems”, International Journal in Foundations of Computer Science & Technology (IJFCST), Vol.4, No.3, May 2014.
[17] TANG Zi-jiao LI Hong-chan. Application of Network Security Management Based on ACL. Journal of Sichuan, University of Science & Engineering (Natural Science Edition), 2009.
[18] ZENG Kuang-Yi, YANG Jia Hai, “Towards the Optimization of Access Control List Journal of Software”. 2007.4:978-985.
[19] SHEN Zhong cheng, “The Application of ACL in Campus Network. Computer Knowledge and Technology” .2008.4:2965-2966.

Authors

Dr.S.S.Tyagi is presently working as a Professor and Head of the Department of Computer Science and Engineering in Manav Rachna International University, Faridabad, Haryana, India. He is having an experience of 22 years including 4 years of industrial and 18 years of teaching experience. He has been holding various academic and administrative positions during his career. He has been consultant to some software development companies. He has been an examiner and evaluator for M.Tech thesis and PhD thesis. He has been a reviewer for books and research papers for some renowned and reputed journals. He is guiding 07 Ph.D. Scholars in the field of Network Security, Ad hoc networks, Cloud Computing, Wireless Security etc. There are around 40 publications to his credit published in reputed International Journals, National Journals and in the proceedings of International and National Conferences and contributing to the research for the benefit of mankind and society at large. His knowledge covers all major areas of Computer Science and Engineering. Currently his areas of research interest are Network Security, Wireless Communication, Mobile Ad hoc Networks, and Cloud Computing. Dr. S. S. Tyagi, is a member of various professional bodies like IEEE, CSI, QCI, ASQ etc.

Kuldeep Tomar is a Research Scholar in the Department of CSE, MRIU, Faridabad, Haryana, India. He has done M.E/M.Tech in Computer Science and Engineering from C.I.T.M., Faridabad, India. He has a total work experience of 12 years (including academics and industry) in different organizations. He is currently working as Associate Professor in NGF College of Engineering & Technology, Palwal, Haryana, Indaia. He has published more than 17 papers in reputed International Journals, National Journals and in the proceedings of International and National Conferences etc. Has is also written a book. He also is a member of Computer Society of India, Membership No: N1039627.