International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Filtering of Malicious Traffic Based on Optimal Source

Pikkili Mahendra[1], K. Raghavendra Rao[2]

[1]M.Tech, Department of CSE, Anurag Group of Institutions, Hyderabad, India

[2]Assistant Professor, Department of CSE, Anurag Group of Institutions, Hyderabad, India

Abstract: ***We have considered the problem of blocking malicious traffic on the Internet via optimal source-based filtering. In particular, we can consider filtering via access control lists (ACLs): These are already available at the routers, but they are a scarce resource because they are stored in the expensive ternary content addressable memory (TCAM). Aggregation (by filtering source prefixes instead of individual IP addresses) helps the less number of filters, but also at the cost of blocking legitimate traffic originating from the filtered prefixes. We have show how to optimally choose which source prefixes to filter for a variety of realistic attack scenarios and operators' policies. In each scenario, we have design optimal, yet to be computationally efficient, algorithms. Using logs from the Dshield.org, We evaluate the algorithms and demonstrate that they bring significant benefit in practice.***

Keywords: Network Security, Internet, Clustering Algorithms, Filtering

1. Introduction

How can we protect our network infrastructure from malicious traffic, such as scanning, malicious propagation, spam, and distributed denial-of-service (DDoS) attacks? These activities cause problems on a regular basis, ranging from the simple annoyance to severe financial, operational, and political damage to companies and organizations, and critical infrastructure. In recent years, they can increased in volume, sophistication, and automation, and also largely enabled by botnets, which are used as the platform for launching these attacks. Protecting a victim (host or network) from malicious traffic is a hard problem that requires the coordination of several complementary components, including nontechnical (e.g., business and legal) and technical solutions (at the application and/or network levels). Filtering support from the network is a fundamental building block in this effort. For an example, an Internet service provider (ISP) may use filtering in response to an ongoing DDoS attack to block the DDoS traffic before it reaches its clients and also Another ISP may want to proactively identify and block traffic carrying malicious code before it reaches and compromises vulnerable hosts in the first place. In both case and filtering is a necessary operation that must be performed within the network. Filtering capabilities are already available at routers today via access control lists (ACLs). ACLs enable a router to match a packet header against predefined rules and take predefined actions on the matching packets [1], and they are currently used for enforcing a variety of policies, including infrastructure protection [2]. For the purpose of blocking malicious traffic was a filter is a simple ACL rule that can denies access to a source IP address or prefix.

To keep up with the high forwarding rates of modern routers, filtering is implemented in hardware: ACLs are typically stored in ternary content addressable memory (TCAM), which allows for parallel access and reduces the number of lookups per forwarded packet. However, TCAM is more expensive and consumes more space and power than conventional memory. The size and cost of TCAM puts a limit on the number of filters, and this is not expected to change in the near future. With thousands or tens of thousands of filters per path, an ISP alone cannot hope to block the currently witnessed attacks, not to mention attacks from multimillion-node botnets expected in the near future.

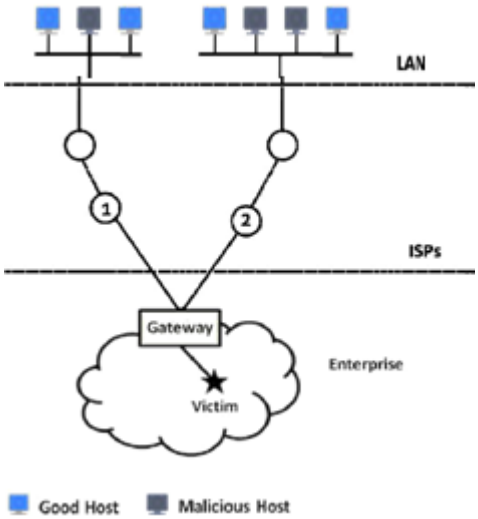

Figure 1: Actual Network

Consider the example shown in Fig. 1: An attacker commands a large number of compromised hosts to send traffic to a victim (say a Web server), thus exhausting the resources of and preventing it from serving its legitimate clients. The ISP of tries to protecting its client by blocking the attack at the gateway router. Ideally it should install one separate filter to block traffic from each attack source, but there are typically fewer filters than attack sources, hence aggregation is used, i.e., a single filter (ACL) is used to block an entire source address prefix. This has been desired effect of reducing the number of filters necessary to block all attack traffic, but also they perform the undesired effect of blocking legitimate traffic originating from the blocked prefixes (We will call the damage that results from blocking legitimate traffic "collateral damage"). Therefore, filter

Licensed Under Creative Commons Attribution CC BY

selection c an be viewed as an optimization problem that tries to block as many attack sources with as little collateral damage as possible, given a limited number of filters. Furthermore, several measurement studies have demonstrated that malicious sources exhibit temporal and spatial clustering [3]–[9], a feature that can be exploited by prefix- based filtering.

In this paper, We proposed a heuristics based solution to stop malicious traffic from internet on to the servers. Our solution is very adaptive to traffic pattern. The filtering are loaded into firewall based on the observation of traffic. These filters achieve black list or white listing of traffic source prefixes.

2. Literature Survey

Fabio Soldo in their paper "Optimal source based filtering of malicious traffic" formulate a general framework for studying source prefix filtering as a resource allocation problem. The best of our knowledge, optimal filter selection has not been explored so far other than Fabio soldo, as most related work on filtering has focused on protocol and architectural aspects. Within this framework, they formulate and solve five practical source-address filtering problems, they depending on the attack scenario and the operator's policy and constraints. His contributions are twofold. On the theoretical side, filter selection optimization leads to novel variations of the multidimensional knapsack problem. He exploited the special structure of each problem and design optimal and computationally efficient algorithms. On the practical side, We have provided a set of cost-efficient algorithms that can be used both by operators to block undesired traffic and by router manufacturers to optimize the use of TCAM and eventually the cost of routers. He used logs from Dshield.org to demonstrate that optimally selecting which source prefixes to filter brings significant benefits compared to non optimized filtering or to generic clustering algorithms [10].Given a set of bad and a set of good source addresses (and), a measure of their importance (the address weights), and a resource budget (plus, possibly, other resources, depending on the particular problem), the goal is to select which source prefixes to filter so as to

Minimize the impact of bad traffic and can be accommodated with the given resource budget. The different variations of the problem can be formulated, depending on the attack scenario and the victim network's policies and constraints: The network operator may want to block all bad addresses or tolerate to leave some unblocked; the attack may be of low rate or a flooding attack; filters may be installed at one or several routers.

In this paper, he formulated five practical filtering problems and developed optimal, yet computationally efficient, algorithms to solve them

2.1 Block-All

Suppose a network operator has a blacklist of size, a white list, and a weight assigned to each address that indicates the amount of traffic originating from that address. The total number of an available filters is Fmax. The first practical goal the operator may have is to install a set of filters that block all bad traffic so as to minimize the amount of good traffic that is blocked. We have design an optimal algorithm that solves this problem at the lowest achievable complexity (linearly increasing with N).

2.2 Block-Some

A blacklist and a white list are given as before, but the operator is now willing to block only some, instead of all, bad traffic, so as to decrease the amount of good traffic blocked at the expense of leaving some bad traffic unblocked. The goal now is to block only those prefixes that have the highest impact and do not contain sources that generate a lot of good traffic, so as to minimize the total cost. We have design an optimal, lowest-complexity (linearly increasing with) algorithm for this problem, as well.

2.3 Time-Varying Block-All/Some

Bad addresses may change over time [4]: New sources may send malicious traffic and, conversely, previously active sources may disappear (e.g., when their vulnerabilities are patched). One way to solve the dynamic versions of BLOCK-ALL (SOME) is to run the algorithms We have propose for the static versions for the blacklist/whitelist pair at each time slot. However, given that subsequent blacklists typically exhibit significant overlap [4], it may be more efficient to exploit this temporal correlation and incrementally update the filtering rules. We have show that is it possible to update the optimal solution, as new IPs are inserted in or removed from the blacklist in time.

2.4 Flooding

In a flooding attack, such as the one shown in Fig. 1,and a large number of compromised hosts send traffic to the victim and exhaust the victim's access bandwidth. In that case, our framework can be used to select the filtering rules that minimize the amount of good traffic that is blocked while meeting the access bandwidth constraint—in particular, the total bandwidth consumed by the unblocked traffic should not exceed the bandwidth of the flooded link.

2.5 Dist-Flooding

All the above problems aim at installing filters at a single router. However, a network operator may use the filtering resources collaboratively across several routers to better defend against an attack. Distributed filtering may also be enabled by the cooperation across several ISPs against a common enemy. The question in both cases is not only which prefixes to block, but also at which router to install the filters. The problem with this solution is that, the weights are provided for the black list and white list users at configuration time and based on the weights alone the optimal set of rules are determined and loaded to firewall. There is no importance to the current traffic. This causes some un hit firewall rules in the firewall memory which would have been used to reduce the damage due to some black list traffic. This motivated me to design a heuristics

Licensed Under Creative Commons Attribution CC BY

solution which takes into account the current traffic pattern. The rules loaded in the firewall are thus very dynamic in our approach. proposed solution is heuristics because the solution gain may not be the maximum but near around to maximum.

3. Overview of Proposed Solution

Our proposed solution consists of two stages.

1. Watching the traffic distribution from the sources and based on it gives score to the traffic sources.
2. Selecting the firewall rules to be loaded into memory based on the traffic distribution score and to maximize the gain of blocking and allowing traffic.

To build traffic distribution at the firewall for each sources every time period, requires some extra resources at firewall. This can be added to firewall and cost is reasonable considering the advantage it brings to the gain of blocking and allowing traffic. This we prove through the simulation and measure the gain achieved due to blocking and allowing traffic. We compare the performance of the proposed solution with BLOCK– ALL solution proposed by Fabio Soldo and prove that We achieve better gain than their approach.

4. Details of Proposed Security Mechanism

4.1 Traffic Distribution Score

To start with We will load the rules with maximum weight on black list into the TCAM memory in firewall. Every time period the firewall builds the traffic distribution vector. The time period is configurable at the firewall. The traffic distribution vector gives that distribution of traffic for the black list and white list prefixes configured in the firewall. The score for each prefixes is then normalized with respect to the total traffic in that time period.

Score of prefix = Count of Hit / Total number of packet received.

4.2 Selection of Firewall Rules

Every time period once after score for the each prefixes are calculated, We need to select the firewall rules to be loaded into the TCAM memory for next time period. To do this We first sort the prefixes based on the score from highest to lowest. We have to select the maximum rules to be put into the TCAM memory as follows:

No_rules_loaded = 0;

For all Prefix in the Blacklist and Whitelist

{

I = Select_Next_High_Score_Prefix(); W(I) = 0;

Rule Set = Find-Associated-Rules(I); For all rules in Rule Set

{

W(I) = W(I) + score of all Prefix Matching rule;

}

}

SPrefixlist Store all Prefix based in weight in descending order

Clear all Rules in TCAM memory While (No_rules_loaded < Max)

{

P = Select_Next_Prefix_from_SPrefixlist;

Rules <- Select_the_rules_for_Prefix(P); Load Rules to TCAM memory; No_rules_loaded = count(Rules);

}

5. Performance Analysis

We used 61-day logs from Dshield.org [10], a repository of firewall and intrusion detection logs collected. The dataset consists of 758 698 491 attack reports, from 32 950 391 different IP sources belonging to about 600 contributing organizations. Each report includes a timestamp, the contributor ID, and the information for the flow that raised the alarm, including the (malicious) source IP and the (victim) destination IP. Looking at the attack sources in the logs, We verified that malicious sources are clustered in a few prefixes, rather than uniformly distributed over the IP space, consistently with what was observed before, e.g., in [3]–[7].

In this simulation, we considered a blacklist to be the set of sources attacking a particular organization (victim) during a single day-period. The degree of clustering varied significantly in the blacklists of different victims and across different days. The higher the clustering, the more benefit We expect from my approach. We also simulated the whitelist by generating good IP addresses according to the multifractal distribution in [16] on routable prefixes. We performed the simulations on a Linux machine with a 2.4-GHz processor with 2 GB RAM.

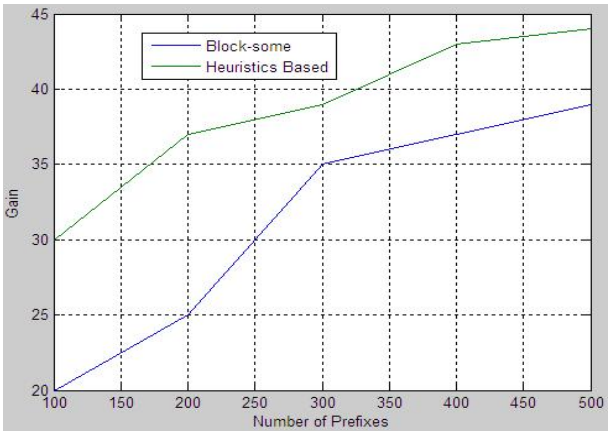

Figure 2: Comparison of Block-Some and Heuristics Based on Gain versus Number of Prefixes.

Licensed Under Creative Commons Attribution CC BY

From the performance analysis, we see that gain achieved in heuristics is much more than of block all.

6. Conclusion

In this paper, we have proposed a heuristics solution to filter malicious traffic. This approach gives better gain compare with previous approach. Every time period once based on the observed traffic rules to loaded are selected and thus it is very adaptive. Currently we are using the weight value arrived for each prefix and totally refresh the TCAM memory but instead weighted scheme like EWMA can be used. Also the process to calculate the weight for each prefix can be done in parallel to reduce the computation time in calculating the weight.

References

[1] "Understanding ACL on Catalyst 6500 series swit ches," Cisco Systems, San Jose, CA, 2003 [Online].Available:http://www.cisco.com/en/US/product s/hw/switches/ps708/products_white_paper091 86a00800c9470.s html.
[2] Protecting your core: Infrastructure protectio n access control lists," Cisco Systems, San Jose, C A, 2008 [Online].Available:http://www.cisco.com/en/US/tech/tk 648/tk361/technologies_white_paper09186a008 01a1a55.shtml
[3] M. Collins, T. Shimeall, S. Faber, J. Janies, R.Weaver,M.De Shon, and J. Kadane, "Using uncleanli ness to predict future botnet addresses," in Proc. ACM Inte rnet Meas. Conf., San Diego, CA, Oct. 2007, pp. 93–104.
[4] Z. Chen, C. Ji, and P. Barford, "Spatial-tempor al characteristics of internet malicious sources," in Proc. IEEE INFOCOM Mini-Conf., Phoenix, AZ, May 2008, pp. 2306–2314.
[5] Z. Mao, V. Sekar, O. Spatscheck, J. Van Der Merwe, and R. Vasudevan, "Analyzing large DDoS attacks using multiple data sources," in Proc. ACM SIGCOMM Workshop Large-Scale Attack Defense, Pisa, Italy, Sep. 2006, pp. 161–168.
[6] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," in Proc. ACM SIGCOMM, Pisa, Italy, Sep. 2006, pp. 291–302.
[7] S. Venkataraman, S. Sen, O. Spatscheck, P. Haffner, and D. Song, "Exploiting network structure for proactive spam mitigation," presented at the USENIX Security Symp., Boston, MA, Aug. 2007.
[8] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber, "How dynamic are IP addresses?," in Proc. ACM SIGCOMM, Kyoto, Japan, Aug. 2007, pp. 301–312.
[9] J. Zhang, P. Porras, and J. Ullrich, "Highly pr edictive blacklisting," presented at the USENIX Sec urity Symp., San Jose, CA, Jul. 2008.
[10] "Dshield: Cooperative network security communi ty: Internet security," Dshield.org [Online]. Available: http://www.dshield.org

Author Profile

Pikkili Mahendra received the B. Tech degree in computer science and Engineering from JNTU Kakinada in 2012 and pursuing M. Tech. degree in Computer science and Engineering from Anurag Group of Institutions, JNTU Hyderabad.

K. Raghavendra Rao working as Assistant Professor in Computer Science Engineering from CVSR College of Engineering from Anurag Group of Institutions Venkatapur(V), Ghatkesar (M), Ranga Reddy District, Hyderabad-500088, Telangana State.

Licensed Under Creative Commons Attribution CC BY