RESEARCH PAPER | Engineering | Volume : 3 | Issue : 4 | April 2013 | ISSN - 2249-555X

A Novel Method to Reduce Source Based Filtering

KEYWORDS: Filtering, aggregation, blocking, ternary

T. Jayanthi	Ms. V. Geetha	A. Kesavan
Assistant Professor, Dept. of Computer Science and Engg. SCSVMV University, Kanchipuram	Assistant Professor, Dept. of Computer Science and Engg. SCSVMV University, Kanchipuram	PG scholar, Dept. of Computer Science and Engg. SCSVMV University, Kanchipuram

ABSTRACT *In this paper, we consider the problem of blocking malicious traffic on the Internet via source-based filtering. Already routers used access control lists (ACLs). ACLs enable a router to match a packet header against predefined rules and take predefined actions on the matching packets. It stored the information in the expensive ternary content addressable memory (TCAM). Aggregation helps to reduce the number of filters. The Experimental result shows that, to improve the performance of filtering method using algorithms*

1. INTRODUCTION

HOW CAN we protect our network infrastructure from malicious traffic, such as scanning, malicious code propagation, spam, and distributed denial-of-service (DDoS) attacks? These activities cause problems on a regular basis, ranging from simple annoyance to severe financial, operational, and political damage to companies, organizations, and critical infrastructure. In recent years, they have increased in volume, sophistication, and automation, 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 level). Filtering support from the network is a fundamental building block in this effort. For 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. 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 either case, 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, a filter is a simple ACL rule that 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.1 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. Consider the example shown in Fig. 1(a): 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 protect its client by blocking the attack at the gateway router . Ideally, should install one separate filter to block traffic from each attack source. However, 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 the desired effect of reducing the number of filters necessary to block all attack traffic, but also 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 selection can 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], a feature that can be exploited by prefix-based filtering. In this paper, we formulate a general framework for studying source prefix filtering as a resource allocation problem. To the best of our knowledge, optimal filter selection has not been explored so far, as most related work on filtering has focused on protocol and architectural aspects. Within this framework, we formulate and solve five practical source-address filtering problems, depending on the attack scenario and the operator's policy and constraints. Our contributions are twofold. On the theoretical side, filter selection optimization leads to novel variations of the multidimensional knapsack problem. We exploit the special structure of each problem and design optimal and computationally efficient algorithms. On the practical side, we provide 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. We use 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.

2. PROBLEM OF FILTERING METHOD

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.1 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.

In this paper, we formulate five practical filtering problems and develop optimal, yet computationally efficient, algorithms to solve them. Here, we summarize the rationale behind each problem and outline our main results; the exact formulation and detailed solution is provided in Section III.

BLOCK-ALL: Suppose a network operator has a blacklist of size N, a whitelist WL, and a weight assigned to each address that indicates the amount of traffic originating from that address. The total number of available filters is Fmax. The first practical goal the operator may have is to install a set of fil-

Volume : 3 | Issue : 4 | April 2013 | ISSN - 2249-555X

ters that block all bad traffic so as to minimize the amount of good traffic that is blocked. We design an optimal algorithm that solves this problem at the lowest achievable complexity (linearly increasing with N).

BLOCK-SOME: A blacklist and a whitelist 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 in (1). We design an optimal, lowest-complexity (linearly increasing with N) algorithm for this problem, as well.

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 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.

3. PROPOSED SYSTEM

In this paper, we formulate a general framework for studying source prefix filtering as a resource allocation problem. To the best of our knowledge, optimal filter selection has not been explored so far, as most related work on filtering has focused on protocol and architectural aspects. Within this framework, we formulate and solve five practical source-address filtering problems, depending on the attack scenario and the operator's policy and constraints. Our contributions are twofold. On the theoretical side, filter selection optimization leads to novel variations of the multidimensional knapsack problem. We exploit the special structure of each problem and design optimal and computationally efficient algorithms. On the practical side, we provide 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.

ADVANTAGES OF PROPOSED SYSTEM:

The proposed system can be used to protect all network infra-structure from malicious traffic, such as scanning, malicious code propagation, spam, and distributed denial-of-service (DDoS) attacks.

3.1. IMPLEMENTATION

To implement the proposed system classified as four modules.

- ✓ Network Creation Module
- ✓ Optimal Source based filtering module
- ✓ Filter Selection Module
- ✓ Evaluation module

Network Creation Module

In this module we construct a network using socket programming, as shown in our Architecture. Where the users can send data to other nodes/network by using the options given. The user node will be listing all the nodes which are connected to the network. The sender can able to select the node name and then send the data.

Optimal Source based filtering module

In this module we design Framework for optimal filter selection

- defined various filtering problems
- designed efficient algorithms to solve them
- Lead to significant improvements on real datasets
- Compared to non-optimized filter selection, to generic Clustering, or to uncoordinated routers
- Because of clustering of malicious sources

Filter Selection Module

In this module we implement the following filter algorithms:

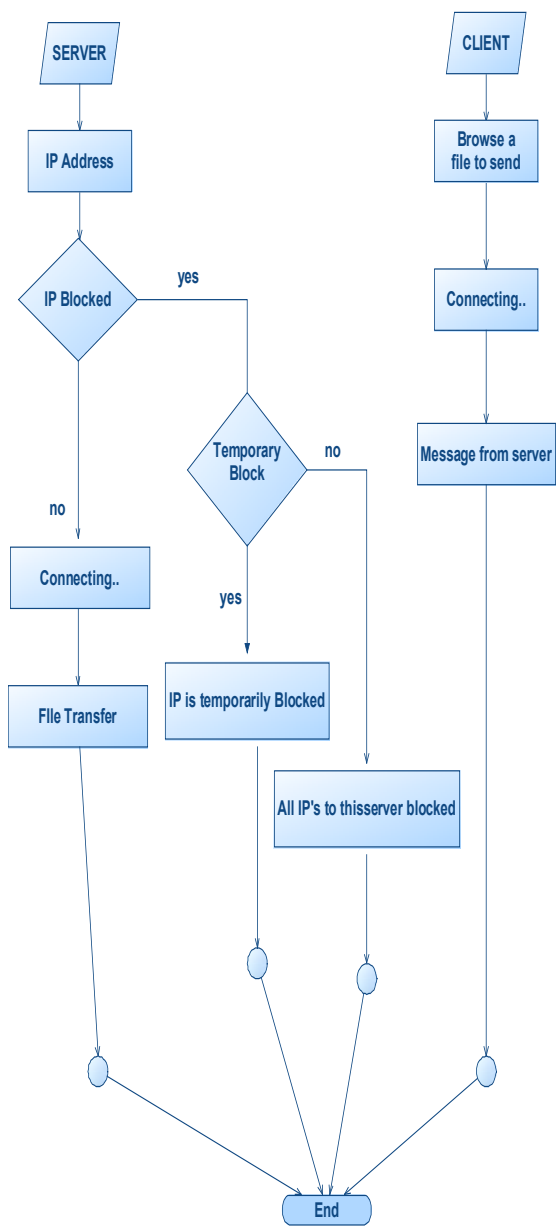
BLOCK-ALL
BLOCK-SOME
TIME-VARYING BLOCK-ALL/SOME

Evaluation module

In evaluation module, the evaluation nodes list the details of the malicious node and the good nodes. This node is designed as such it will be refreshed for a few seconds of period to update the information on each and every second. This node acts as a evaluation node as since it evaluates the nodes from malicious ones.

3.2. SYSTEM FLOW DIAGRAM

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.



Volume : 3 | Issue : 4 | April 2013 | ISSN - 2249-555X

4. EXPERIMENTAL RESULT

4.1 SCREEN SHOTS

5. CONCLUSION

In this paper, we introduce a framework for optimal source prefix-based filtering. The framework is rooted at the theory of the knapsack problem and provides a novel extension to it. Within it, we formulate five practical problems, presented in increasing order of complexity. For each problem, we designed optimal algorithms that are also low-complexity (linear or pseudo-polynomial in the input size). We simulate our algorithms over Dshield.org logs and demonstrate that they bring significant benefit compared to non-optimized filter selection or to generic clustering algorithms. A key insight behind that benefit is that our algorithms exploit the spatial and temporal clustering exhibited by sources of malicious traffic.

REFERENCE [1] "Understanding ACL on Catalyst 6500 series switches," Cisco Systems, San Jose, CA, 2003 [Online].Available:http://www.cisco.com/en/US/products/hw/switches/ps708/products_white_paper09186a00800c9470.shtml | [2] "Protecting your core: Infrastructure protection access control lists," Cisco Systems, San Jose, CA, 2008 [Online]. Available: http://www.cisco.com/en/US/tech/tk648/tk361/technologies_white_paper09186a00801a1a55.shtml | [3] M. Collins, T. Shimeall, S. Faber, J. Janies, R.Weaver,M.De Shon, and J. Kadane, "Using uncleanliness to predict future botnet addresses," in Proc. ACM Internet Meas. Conf., San Diego, CA, Oct. 2007, pp.93–104. | [4] Z. Chen, C. Ji, and P. Barford, "Spatial-temporal characteristics of internet malicious sources," in Proc. IEEE INFOCOM Mini-Conf.,Phoenix, AZ, May 2008, pp. 2306–2314. | [5] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," in Proc. ACM SIGCOMM, Pisa, Italy, Sep. 2006, pp. 291–302. | [6] 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. | [7] J. Zhang, P. Porras, and J. Ullrich, "Highly predictive blacklisting," presented at the USENIX Security Symp., San Jose, CA, Jul. 2008. | [8] "Dshield: Cooperative network security community—Internet security," Dshield.org [Online]. Available: http://www.dshield.org