**Optimal Source-Based Filtering of Malicious Traffic**

ABSTRACT:

In this paper, we consider the problem of blocking malicious traffic on the Internet via source-based filtering. In particular, we consider filtering via access control lists (ACLs): These are already available at the routers today, but 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 reduce the number of filters, but comes also at the cost of blocking legitimate traffic originating from the filtered prefixes. We show how to optimally choose which source prefixes to filter for a variety of realistic attack scenarios and operators’ policies. In each scenario, we design optimal, yet computationally efficient, algorithms. Using logs from Dshield.org, we evaluate the algorithms and demonstrate that they bring significant benefit in practice.

**EXISTING SYSTEM:**

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.

**DISADVANTAGES OF EXISTING SYSTEM:**

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.

**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.

MODULES:

ü Network Creation Module

ü Optimal Source based filtering module

ü Filter Selection Module

ü Evaluation module

HARDWARE REQUIREMENTS:

• System : Pentium IV 2.4 GHz.

• Hard Disk : 40 GB.

• Floppy Drive : 1.44 Mb.

• Monitor : 15 VGA Colour.

• Mouse : Logitech.

• Ram : 512 Mb.

SOFTWARE REQUIREMENTS:

• Operating system : - Windows XP.

• Coding Language : VB.NET