High Performance Traffic Shaping for DDoS Mitigation

Markus Goldstein, Matthias Reif, Armin Stahl
German Research Center for Artificial Intelligence DFKI GmbH
Research Group Image Understanding and Pattern Recognition (IUPR)
D-67663 Kaiserslautern, Germany
{goldstein,reif,stahl}@iupr.dfki.de

Thomas Breuel
Technical University of Kaiserslautern
Department of Computer Science
Research Group Image Understanding and Pattern Recognition (IUPR)
D-67663 Kaiserslautern, Germany
breuel@iupr.dfki.de

ABSTRACT

Distributed Denial of Service (DDoS) attack mitigation systems usually generate a list of filter rules in order to block malicious traffic. In contrast to this binary decision we suggest to use traffic shaping whereas the bandwidth limit is defined by the probability of a source to be a legal user. As a proof of concept, we implemented a simple high performance Linux kernel module *nf-HiShape* which is able to shape thousands of source IP addresses at different bandwidth limits even under high packet rates. Our shaping algorithm is comparable to *Random Early Detection (RED)* applied on every single source IP range. The evaluation shows, that our kernel module can handle up to 50,000 IP ranges at nearly constant throughput whereas Linux *tc* already decreases throughput at about 200 ranges.

1. INTRODUCTION

From a machine learning point of view, mitigation of Distributed Denial of Service (DDoS) attacks is a fairly easy task as long as attacking sources send malicious packets with identifiable features. Many researchers worked in this area and succeeded in the mitigation of SYN, ICMP or UDP floods near target [6]. If attackers do not violate protocols and generate TCP flows which are not distinguishable from normal flows but occur in very high numbers, mitigation is a challenging task. In this case, statistics of flow features can be used to compute a *Conditional Legitimate Probability (CLP)* [5] of packets from a source to be legal traffic. A CLP threshold then defines if a source is accepted or denied.

We suggest to apply traffic shaping of the single sources instead of the binary accept/drop decision leading to a better use of the CLP information. Sources with a high CLP will get more priority and bandwidth during an attack than sources which are likely part of the attacking bot network. This will lead to less *collateral damage* (denying legal users) and a better usage of the available resources.

ACM CoNEXT 2008 Student Workshop, December 9, 2008, Madrid, SPAIN

2. TRAFFIC SHAPING

For practical traffic shaping, *tc* [2] is typically used on Linux systems or *ALTQ* on BSD. These tools are highly configurable but they store matching rules for packet classification in a linear list. If there are hundreds or thousands of sources defined, each arriving packet has to traverse the list from top to bottom. This access time latency is especially non-productive in a DDoS incident.

The approach we introduce in the following determines the bandwidth limit of a flow by its source IP address. An IP range is a continuous interval r over IP addresses $[r^{start}, r^{end}]$ with a defined bandwidth limit. The set of ranges $R = \{r_o, ..., r_n\}$ allows entries of arbitrary length but permits ranges that overlap. Now we can sort the set of ranges:

$$\forall r_i, r_j \in R, i < j : r_i^{end} < r_j^{start} \quad (1)$$

If there is no range available for an IP address, the packet will pass the shaping procedure without being queued or dropped (no limit). Due to the sorted list the bandwidth limit corresponding to an incoming packet can be determined fast by performing a binary search with a complexity $O(\log_2 n)$, which is crucial in the DDoS mitigation scenario. In the worst case for each single IPv4 address a different bandwidth limit is assigned ($n = 2^{32}$) leading to a maximum of 32 lookups for each incoming packet. This packet classification problem has also been addressed by Feldmann et. al. [3] proposing a FIS tree with a better look-up complexity of $O(\log_2 \log_2 N)$ where N is the total range of IPv4 addresses. Another advantage of this approach is the fact that more than one IP packet feature can be used for classification. However, it requires complex datastructures and thus a lot of time to create the FIS tree and

therefore we neglect this approach in our early research stage. We evaluate later in Section 4 our approach and show that it performs good enough with many rules in order to mitigate even very highly distributed DoS attacks.

3. SHAPING ALGORITHM

The presented efficient shaping algorithm ensures the applicability at even very high packet rates. It consists of mainly two procedures: The first one is applied at every packet arrival and only decides if the packet should be accepted, dropped or queued similar to *Random Early Detection (RED)* [4]. The second function will be called in a constant time interval. It resets the bandwidth counters of a range and tries to send the least recent packets of the queues as long as their limits are not reached. Pseudo code for both functions is given in Algorithm 1.

Algorithm 1 nf-HiShape

```
 1: function PACKET_HANDLER(Packet p)
 2:     r ← range incl. p.source_IP        ▷ binary search
 3:     if r not found then
 4:         accept(p) and return
 5:     q ← queue of r
 6:     if not q.empty or r.sent+p.size > r.limit then
 7:         if q.size < q.max_size then
 8:             q.push(p)
 9:             steel(p)
10:         else drop(p)
11:     else
12:         r.sent + = p.size
13:         accept(p)
14: function TIMER_HANDLER
15:     for all ranges r do
16:         r.sent ← 0; finished ← false
17:         q ← queue of r
18:         while not q.empty and not finished do
19:             p ← q.front()
20:             if r.sent + p.size < r.limit then
21:                 send(p)
22:                 q.pop()
23:                 r.sent += p.size
24:             else finished ← true
```

4. EVALUATION

We compare the presented approach with *tc* using ingress shaping to low bitrates. The performance of both methods is measured by the throughput of a legal and not limited user on a 1GBit link. In Figure 1 it is shown that the throughput of *tc* significantly decreases at a certain amount of ranges and drops to almost zero, especially for small packets (MTU). The throughput of *nf-HiShape* stays almost constant and drops only slightly on very large range sets.

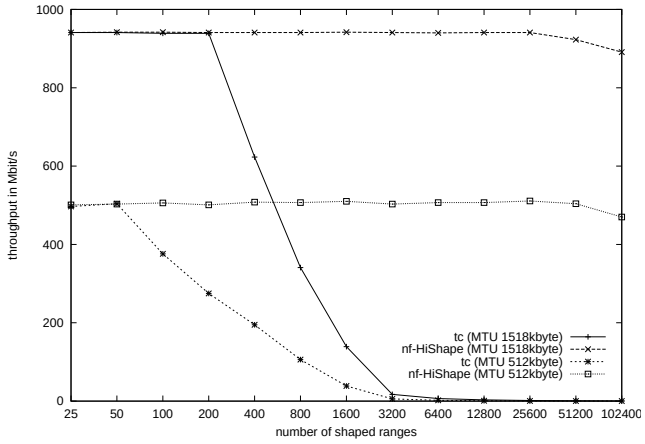

Figure 1: *nf-HiShape* is able to handle more traffic than *tc* on large sets of shaping rules

5. CONCLUSION AND FUTURE WORK

In this paper we presented an efficient algorithm to shape the bandwidth usage of many source IP addresses individually. The evaluation showed that the presented approach *nf-HiShape* outperforms *tc* using a large number of source ranges and is therefore more suitable for DDoS mitigation. Our next steps include investigation and implementation of more powerful packet classification methods as proposed in [3] in order to support multiple packet attributes.

The presented algorithm was implemented as a Linux kernel module and is available as an open source release [1] including a userland tool.

6. REFERENCES

[1] nf-hishape. http://code.google.com/p/nf-hishape/.

[2] W. Almesberger. Linux Network Traffic Control – Implementation Overview. In *5th Annual Linux Expo*, pages 153–164, 1999.

[3] A. Feldmann and S. Muthukrishnan. Tradeoffs for packet classification. In *INFOCOM*, pages 1193–1202, 2000.

[4] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1:397–413, 1993.

[5] Y. Kim, W. C. Lau, M. C. Chuah, and H. J. Chao. Packetscore: A statistics-based packet filtering scheme against distributed denial-of-service attacks. *IEEE Transactions on Dependable and Secure Computing*, 03(2):141–155, 2006.

[6] J. Mirkovic, S. Dietrich, D. Dittrich, and P. Reiher. *Internet Denial of Service: Attack and Defense Mechanisms (Radia Perlman Computer Networking and Security)*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.