Denying Denial-of-Service Attacks: A Router Based Solution[1]

Shu Zhang
Arizona State University
Tempe, AZ 85287
shu.zhang@asu.edu

Partha Dasgupta
Arizona State University
Tempe, AZ 85287
partha@asu.edu

Abstract

Distributed Denial-of-Service (DDoS) attacks prevent users from accessing services on the target network by flooding the target network with a large volume of traffic.

In this paper, we propose a "Hardened Network" system, which is based on intelligent routers. This network can be incrementally deployed on the Internet and can be used to detect, stop, and recover from DDoS attacks. This Hardened Network does not require any modification to the end-systems, such as the client and server hosts. It can detect a DDoS attack before it severely slows down the target machine or the network. Then, it can selectively drop packets close to the sources and hence stop the attack at points that are closer to the attack origin, and continue to provide service.

Key words: DDoS, Autonomous System, Cryptography

1. Introduction

DDoS attacks have become a serious threat since the SYN attacks flooded and shut down several Web Servers in September of 1996. Due to the automated tools and open source environment, DDoS attacks can now be easily launched simultaneously from thousands of compromised systems. More recently, a number of DDoS attacks have been launched on various Internet sites including some of the Internet News and Information sites, such as Foxnews.com, ESPN.com, and ABCNEWS.com.

A Denial-of-Service (DoS) attack is characterized by "an explicit attempt by an attacker to prevent legitimate users of a service from using that service" [1]. The Distributed Denial-of-Service (DDoS) attack is a notorious extension of the DoS attack**.** A DDoS attack is launched by flooding a large number of attack packets to a target machine, with the simultaneous collaboration of hundreds or thousands, or even more computers that are scattered all over the Internet. The attack traffic consumes the resources of the Network or the target machine, so that the legitimate requests will have to be discarded due to the lack of resources for either transportation or processing, such as bandwidth and receiving buffer at the server end.

DDoS attack packets are normal-looking packets, therefore, it is hard to prevent or detect this type of attacks. At the current stage, DDoS attacks can only be detected when the target machine performance is severely degraded, or completely shut down, or the network close to the target is tremendously congested. Furthermore, source address spoofing and stateless network routing make the traceback of such attacks non-trivial.

Several DDoS defense systems have been proposed. Most of them augment the Internet devices with monitoring components to detect attacks, trace-back components to trace the sources of the attacks, and rate-limiting components to recover from the attacks. These Internet devices may be located at the victim network, the intermediate network, or the source network [4]. The problem with these approaches is that they require the Internet devices to be aware of the defense system, or the topology of network, or at least, the upstream network. In addition, they might require the cooperation of all the devices in the intermediate network, or the updating of most of the Internet devices.

In this paper, we present the DDoS defense system, which builds protection at the transport layer of the Internet. We propose the upgrading of selected routers to what we call "Hardened Routers." Hardened Routers are routers augmented with capabilities of encryption, signing, verifying, and dropping the packets that they route. These functions gives right to enable flow control, privacy, source authentication, undeniability, and enhanced (and secure) global network control. There is *no* change to the protocols and protocol stacks at the hosts. Thus, the applications do *not* have to be changed or be configured. All TCP/IP enabled devices benefit from the upgraded network.

The Hardened Routers are completely compatible with the existing routers, and can be *incrementally* introduced – making deployment quite feasible. The addition of the intelligent, security capable routers is called "*hardening*" the

[1] This research is partially supported by grants from AFOSR, DARPA, and NSF.

network. By replacing regular network routers with Hardened Routers, the infrastructure of the network is hardened, mainly because the Hardened Routers can perform a few extra functions in addition to routing. The Hardened Routers do not need to collaborate with each other until an attack is detected**,** while the cooperation involves only the routers that are located at both ends of each attack path. We do not assume that Hardened Routers know the topology of the entire network, or even the topology of the Hardened Network. (Note: The Hardened Network is a subset of the entire network).

2. Related Work

Many techniques have been proposed against DDoS attack. In this paper, we focus mainly on DDoS detection techniques, such as MULTOPS [12], and D-WARD [5]; and traceback techniques, such as IP Traceback [8], ICMP Traceback [9], and CenterTrack [3]. Pushback [4] integrates both techniques.

The MULTOPS [12]-equipped router uses the disproportion of the two rates: to-rate and from-rate as a heuristic to detect attacks. MULTOPS has to be combined with other techniques to trace the origin and response to the attack.

D-WARD needs to be deployed at the network entry points to prevent the systems in the source-end network from participating attacks. It detects the sign of DDoS attack by monitoring the two-way traffic and comparing it to the normal traffic model. Traceback is easy due to D-WARD is at entry point. Rate-limitation is used to throttle the attack traffic. Deployment could be problematic since it is the victim who feels the direct benefit, not the deploying networks.

ICMP Traceback [9] uses router-generated ICMP traceback messages at a low probability. Victim uses the ICMP Traceback messages to reconstruct the attack paths. The traceback message relies on the router's ability to associate a packet with the input MAC address on which it is received (input debugging). This ability is not available on all the routers. The ICMP traceback message can be dropped when the network is congested. IP Traceback [8] requires routers to mark packets with the router's information at a low probability. The victim can reconstruct the paths if enough packets are marked. Both techniques require sufficient routers to support and enough traceback packets to be received. In addition, both suffer from the faked traceback information and involve some form of authentication. They also fail to stop the attacks.

CenterTrack [3] is an overlay network, which uses IP tunnels to reroute interesting packets to special tracking routers. The CenterTrack network is deployed inside an ISP network, and it can only trace back inside the local network. It also requires the Input debugging and dynamic routing support. All traceback techniques do not detect attacks.

Pushback [4] augments routers with a Pushback daemon. The routers log the number of dropped packets due to queue restriction and send them to the daemon. The daemon analyzes them and looks for signs of DDoS attack. It then notifies the local rate-limiter to drop the bad traffic and the daemon of the upstream routers. Pushback requires the routers' support to perform Input debugging. Pushback messages need some form of authentication.

3. Hardened Network

3.1. Overview

To illustrate the Hardened Network, consider the network shown in Fig. 1.

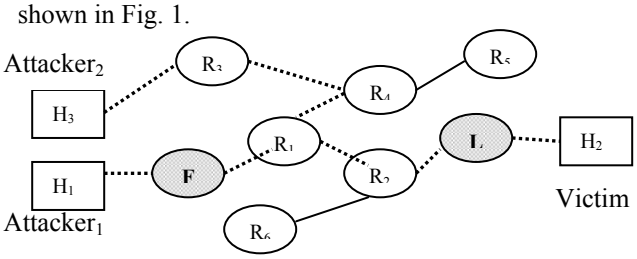

Figure 1. Connections in Hardened Network

Router F and L are Hardened Routers. F is the first hardened router of the path from attackers and the victim, which means there is no hardened router between the $attacker_1$ and F. L is the last Hardened Router, which means there is no Hardened Router between L and the victim.

We assume that in this network only F and L are hardened. The dotted line indicates the attack traffic. When F forwards the packets, it has to attach its non-deniable identification. During the attack, L senses the congestion and the packet dropping due to the queue limitation. According to the identification information, L traces the attack traffic from $attacker_1$ back to F and from $attacker_2$ back to itself. L informs F about it. Both L and F start dropping attack traffic. The attack traffic from $attacker_1$ can be stopped closer to the source, but that not from $attacker_2$. When more routers are hardened, we can stop the attack more efficiently. Even in this network, we still reduce the load to victim, so that it can serve more good traffic. Hence, we can say that only two hardened routers present can ensure that the traffic flowing through them is resilient to attacks (or is protected from attacks). Due to the certificate and signature, the command from L to drop the traffic can not be forged or be denied.

Also we could let F encrypt the packet before it attaches the identification, and let L decrypt the packet. In this way, we could also ensure the privacy between the hardened routers without installing VPN software.

3.2. Architecture

The Internet is a set of Autonomous Systems (AS). The Hardened Network hardens the Internet by replacing the

border/Access routers in an AS with hardened border/Access routers. We call the AS with hardened routers hardened AS.

The key idea of "hardening" is the enhancement of some functions in a router. This enhancement is cryptography-based, including signature and authentication. We take the overhead reduction into account, so instead of digital signature, we use symmetric encryption (see section 3.3).

Cryptographic techniques permit the construction of a trusted subsystem with distributed reach. Distributed reach can span geographically distributed network elements belonging to different administrative domains. Therefore, distributed reach is crucial for detecting attacks, recovering quickly from attacks, and identifying the attackers. Recovery is accomplished using router-oriented means (such as choking off selected attacker traffic).

To facilitate global flow control and manage router configurations and certificates for each hardened AS, we add a central "Hardened AS Controller." Each hardened AS Controller communicates with the neighboring AS controllers to establish a "control network." The control network is a logical, overlay-network that is used for global flow control, certificate issuance, and attack monitoring.

We classified all the routers in an AS into 3 groups: Border Routers, Access Routers, and Core Routers [10]. Border routers manage the traffic between ASes. Access routers are responsible for the incoming and outgoing traffic of the hosts. The rest of routers are core routers. A Hardened AS composition is shown in Fig. 2. We only hardened border routers and access routers.

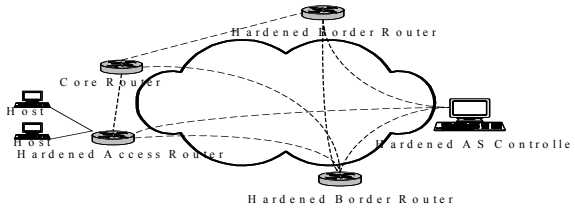

Figure 2. A Hardened AS

Since the Hardened Network can be deployed gradually and incrementally, the hardened ASes co-exist with the conventional ASes, illustrated in Fig. 3.

3.3. First and Last Routers

Due to this co-existence situation, the architecture of the network has some issues. These are:

- Determining when to encrypt and sign the packets and how to ensure low overhead signatures;
- Determining when to decrypt the packets and strip off the signature so that the destination can recognize the packets;
- Performing a key exchange operation;

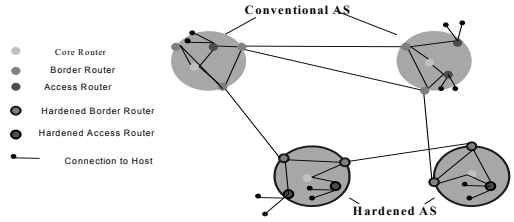

Figure 3. Conventional and Hardened ASes

We still use the network shown in Fig. 1 to describe how we solve these issues. The first hardened router, *F,* will encrypt and sign all the packets of this connection, while all the Hardened Routers on the route, including *F,* will sign the packets. When the last hardened router, *L,* gets the packets, it will strip off all the signatures and decrypt the packets so that H_2 can recognize the packets.

The signatures provide source and route authentication. Adding some information gathering and traffic control in hardened router will allow us to provide attack resilience, especially from distributed attacks such as DDoS.

Although the signatures of all the Hardened Routers on the path gives the more confidence about the origin of packets, this procedure will dramatically decrease the performance, and in order to trace the origin of an attack, the first hardened router will be the most useful part. Hence we decided to require only the first hardened router to sign the packets. To further optimize the performance of our Hardened Network, we chose to perform the symmetric encryption only once, instead of performing the encryption once and then signing it.

Symmetric encryption means that the first hardened routers that encrypt the packets must know the last hardened router and the shared secret before hand. The first hardened router encrypts the payload plus one byte of its identification using the secret and then attaches its identification. The last hardened router retrieves the secret according to the identification; and decrypts the packets and matches the byte with the identification before forwarding them.

Hardened routers get to know the last hardened router on each path by Hardened Border Gateway Protocol (HBGP). HBGP is a modified BGP protocol, which is also compatible with BGP. All the hardened border routers run the HBGP, instead of the BGP. When hardened border routers exchange the BGP routing information, they also attach the hardened network information. The key is shard by the Hardened Routers with the Hardened AS Controller of the last hardened router. Further details about this protocol are included in [10].

As stated earlier, the replacement of regular routers with hardened routers can be done incrementally. A hardened AS (an AS with all the routers hardened) is instantly more resilient to attacks. This resilience increases rapidly as more and

more ASes are upgraded. Any ASes that have not been upgraded remain unmodified and continue working in their conventional way. However, they still benefit from the presence of the upgraded ASes, since the Hardened ASes enhance the overall resilience of the Internet (DDoS resilience).

3.4. Benefit

Consider two hardened Ases A_1 and A_2. When traffic flows between hosts connected to A_1 and A_2, the packets are encrypted and signed at the first hardened access/border router and then decrypted at the last hardened access/border router between the two hosts. This provides three benefits: Privacy, DoS prevention, and other attack prevention.

Privacy: Since the traffic flow between the first and last hardened routers is encrypted, regardless of the path taken by this traffic (including passing through non-hardened or untrusted nodes), the packets are immune to sniffing.

DoS Prevention: If the hosts on A_1 start attacking the hosts on A_2, the routers at A_2 can detect the attack, identify the source hardened routers emanating the traffic and inform the AS Controller of A_1. The AS Controller can then start packet filters on the hardened routers to stop the suspected packets from flooding the destination.

Other Attacks: Similar to DoS prevention, Distributed DoS (DDoS) prevention can be achieved if there are many hardened ASes and the attack traffic flows mainly from or via the hardened ASes. Other attacks that exploit vulnerability can be detected and stopped by stateful filtering at the last router.

The presence of just two hardened ASes provides a number of benefits in the network. Traffic that flows though these ASes can utilize the attack resilience of these ASes even though the end points are not hardened. As the number of hardened ASes grows, the network becomes less vulnerable to attacks.

4. Detecting DDoS

4.1. Hardened Border Router

The routing table of a hardened border router includes the information about the Hardened Network information and four other fields: TRF, TRL, BAD, and GOOD number. BAD and GOOD numbers will be discussed in the recovery process in Section 5. Each entry looks like the following:

(destination, next hop, last Hardened Router IP, TRF, TRL, BAD, GOOD)

In the table in which the decryption keys are stored, each entry is as follows:

(Hardened Router IP, decryption Key, HRF)

The HRF number is used to save information about how many times the decryption key is used, which also means saving information about how many times the corresponding hardened router is the first hardened router on the path.

Hardened border router increments the TRF number whenever it decides to encrypt a packet and increments the TRL number whenever it decrypts a packet. In case that there is no attack, the hardened border router does not update the HRF number; when an attack is detected it has to increment the corresponding HRF number every time that it retrieves the decryption key.

Hardened border router also needs to log the number of dropped packets to any destination due to the queue restriction.

We assume that the information gathering period is μ. So normally every μ time, hardened border router collects the TRF and TRL numbers with the destination address and the dropping rate and sends to the local Controller. If the Controller detects an attack, it notifies all the local hardened routers. At the end of the next μ time, the local hardened border routers also need to report the HRF number and the corresponding first hardened router, along with all the other numbers they report under normal circumstances.

4.2. Hardened Access Router

Hardened access routers only need to be responsible for the traffic that originates from or to the destination of the hosts that are directly connected to them.

When an access router is set up, each hardened access router downloads a list, which includes all the remaining hardened access routers in the same AS, and all the host connection information. The local Controller informs the existing access routers to update their list with this new router.

The hardened access router encrypts only the packets from the connected hosts, whose destinations are in the same AS, and increments the TRF number. Otherwise, the hardened access router simply forwards them and lets the hardened border router encrypt them. The hardened access router also increments TRL when it decrypts a packet. The hardened access router also needs to report these numbers to the local Controller.

4.3. Hardened AS Controller

In addition to the central management of each Hardened AS [10], a Hardened AS Controller has to detect any possible DDoS attacks and respond to them. The components in a Hardened AS Controller used for detecting DDoS are the Monitor Process and the Response Process.

4.3.1. Monitor Process for DDoS Detection

The Monitor Process gets the traffic statistics information from the local hardened routers. It compares these numbers with the normal traffic pattern of each destination.

In case of a DDoS attack, the normal traffic pattern includes the normal TRF/TRL traffic pattern and the normal dropping rate. This information comes from the self-learning of the local Hardened AS Controller, from information collection, and from history events, when the Network environment is stable, which means that, there is no attack, and also no major change to the routing information.

During each μ time, the Monitor Process compares TRL and the dropping rate to the normal traffic pattern. We use the normal distribution to find out if the current TRF and the dropping rate are normal. We use TRL as an example. AVG is the average of TRL; STDEV is the Standard Deviation of the same interval; and Z(x) refers to the possibility of the distribution of the traffic in the area (AVG, AVG + x * STDEV). Suppose the Rule base sets x = α, and constant p (< 1-Z(α)), then the formula is as follows (for the dropping rate, the formula is similar):

```
normal = true;  aggre = 1;
if ( (TRL – AVG)/STDEV< α) {
   normal = true;  aggre = 1;
}
else if ( (TRL – AVG)/STDEV) >= α)
 if ( normal) {
   normal = False;
  aggre =aggre * (1 - z( (TRL – AVG)/STDEV));
   if (aggre <= p)  trigger the Recovery Process;
   repeat the formula till next interval;
 }
 else  {
  aggre =aggre * (1 - z( (TRL – AVG)/STDEV));
   if (aggre <= p)  trigger the Recovery Process;
   repeat the formula till next interval;
 }
```

After the Response Process is triggered, the Monitor Process continues to collect information from the local Hardened Routers, and informs the Response Process about the current status of traffic.

The decision about the predefined μ value is hard. If μ is too large, we can not detect the attack in time, otherwise, it involves too much communications between Routers and Controllers. Therefore we decided to have the Controller ask all the hardened routers to decrease μ, when it detects an attack. In this way, we can define a relatively larger μ value initially.

4.3.2. Response Process for Traceback

After the Response Process is triggered by the local Controller, it gets the HRF numbers and the corresponding first hardened router information from all the local hardened routers.

The Response Process uses this information to trace the first hardened routers. After monitoring this information for couple of μ time periods, it decides which hardened routers are really the first hardened routers on the attacking paths. Through an SSL connection, it notifies the Response Processes of all the remote Hardened Ases which contain these first hardened routers.

5. Recovery for DDoS

When the Response Process of the remote Hardened AS Controller gets the authenticated command to drop traffic, a simple reaction would be to require all the hardened routers to drop all the packets to the target with the same Transportation Layer Protocol. This is the approach that we have simulated (see section 6).

A better approach would be to watch for packets coming from the target. In this approach, we assume that if there is traffic coming from the target, the possibility that the destinations of the traffic are not part of the attack is high.

The hardened router drops each packet going to the target and increments the BAD number. For each packet coming from the target, the hardened router increments the GOOD number in the routing table entry for the destination of the packet. For the rest of the packets, it performs the usual processes..

At the end of μ time, the hardened router reports all the numbers, TRF, TRL, GOOD and BAD to the Controller. It also checks all the non-zero GOOD values, builds a table of allowed IP addresses, and allows the traffic from these addresses to reach the target (and drops all the other packets). This solution allows some degree of QoS at the client while the server is under attack, and reduces the severity of the attack.

In the case of remote Hardened ASes, if the BAD value is considerably lower and the TRF value is a proportion of the total of GOOD numbers, then it takes this as a sign of the end of the attack. It restores the traffic. After restoration, if some notifications are sent from the detecting Hardened AS, the controller has to start dropping packets, or otherwise it can work normally.

6. Measurements

We simulated the Hardened Network by using Scalable Simulator Frameworks (SSFnet) on Intel Pentium III 1GHz, Windows 2000 machines. Fig. 4 shows the topology of the simulated network.

Each AS has only one router, which works as a border router. The Server located at AS12 is illustrated in Fig. 4 in the form of a bold solid circle. While, the Client Ases are

illustrated in the form of dotted circles. Apart from the border router in these ASes, there is also a host machine.

As shown in Fig. 4, the Server ran the HttpServer, and the clients ran the HttpClient. Each AS had only one router, which ran the BGP4 protocol. We configured all the edge routers to be hardened border routers. To simplify the simulation, we implemented the basic functions of the Hardened Controller, such as the Monitor Process and the Response Process, as the components of the hardened border router, not the controller's.

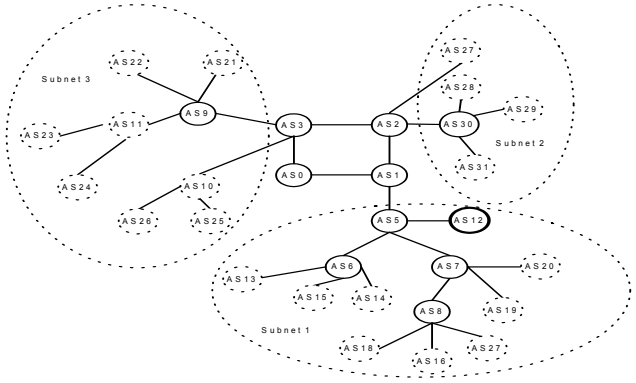

Figure. 4 Topology of the Simulated Network

The simulation lasted 1000 seconds, and the attack traffic was started at the 150th second and ended at the 800th second. The attack traffic that we simulated was the ICMP ECHO Response traffic, which was sent out by several Httpclients in the three subnets to the only Httpserver. Fig. 4 does not differentiate the normal HttpClients and the attack Clients. Each attacker sent out one ECHO Response message every 0.1second. During this simulation, the interval μ was 100 seconds.

Fig. 5 shows the detection of the ICMP attack at the Hardened Border router, in AS12, which is the last Hardened Border Router that is connected to the target Server. The Y axle of Fig. 6 indicates the number of packets arriving at the target Server, while the X axle represents the simulation time.

As shown in Fig. 5, the Monitor Process of the Hardened Border router detected the attack at the 200th second and triggered the Recovery Process at the same time as the line, which is labeled as "detected" in Figure 6. During the detection process, we chose p = 0.0001, which means that x = 3.9 for Z(x), and AVG = 1110, and STDEV = 540.65.

The Response Process analyzed the first hardened router information in the IP packets during the next collection period. It sent out the notifications for dropping the packet to the first hardened router, at the 300th second. In this simulation, we statically set the parameter p in the Recovery process to be 3. This p number can also be dynamically changed according to the x value in Z(x) which is used for the detection. During this simulation, the first hardened router made the first drop of packets to the target at the 300.10000264th second, after receiving the notification. This event is demonstrated by the first line in Fig. 5, which is labeled as "notify" and "drop."

The Monitor Process kept monitoring the traffic. At the 400th second, it still detected the attack**.** Apart from the notifications sent at the 300th second, the Response Process sent out additional notifications to the first hardened routers. In this case, the first drop was made at the 400.00000584th second, and is shown in Fig. 5 as the second line, which is labeled as "notify" and "drop."

In the 500th second, the Monitor Process and the Response Process performed the same procedure, while the drop point started at the 500.00000744th second, which is illustrated in Fig. 5 by the third line of "notify" and "drop."

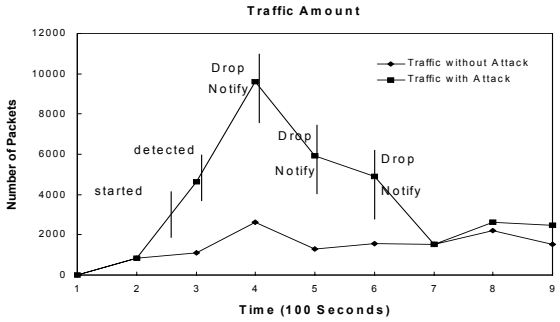

Figure. 5 Traffic Pattern at Server of AS12

Because we hardened all the edge routers, the simulation shown in Fig. 5 successfully detected and responded to the DDoS attack. All the traffic in this simulation, normal or attack, had to go through the only router directly connected with the target, such as the router in AS12.

What if only partial traffic to the target passes through the Hardened Routers? Can the Hardened Network detect and respond to the attack? In order to answer these questions, we conducted another simulation.

In the second simulation, the topology of the network was the same as the one shown in Fig. 4. We hardened the routers in AS0, AS1, AS2, and AS3. The target server was still in AS12. In this simulation, the router in AS1 was the last hardened router for the traffic from subnet 2 and 3, and this router was called router 1. However, in the case of the traffic from subnet 1, the Hardened Network would not monitor it. The result is shown in Fig. 6. The Y axle demonstrates the total number of packets getting to the Server and the number of packets passed through Router 1, while the X axle shows the simulation time.

The attack still started at the 150th second. Router 1 detected the attack at the 200th second. The first notification for dropping the packets was send out by router 1 at the 300th

second. The first drop by the hardened router was made at the 300.00000744th second, illustrated by the first "drop & notification" line. Since router 1 still detected the attack in the next report period, it sent out the notification at the 400th second. The drop for this round was made at the 400.00000584th second, as illustrated by the second "drop & notification" line in Fig. 6.

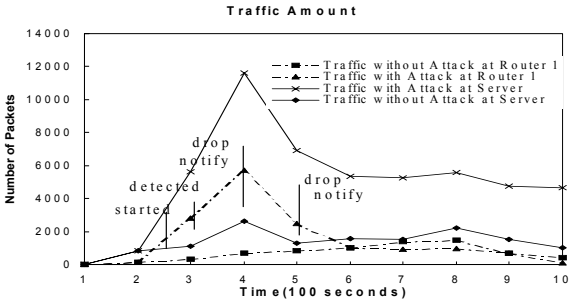

Figure. 6 Traffic Pattern at Router 1 of AS1 and Server of AS12

From Fig. 6, we can see that after router 1 detected the attack and responded to it while the target system was still under the attack, as shown by the line at the top of the figure. This is because the Hardened Network missed the traffic that originated from subnet 1, while some attack agents were located in subnet 1. Although router 1 could not monitor and block the attack traffic from subnet 1, the Hardened Network did reduce the traffic to the target.

7. Ongoing Research

As part of the network-hardening project, we are building prototypes of the actual intelligent routers and we will be deploying them, albeit in a small scale, in our lab. The first prototype uses Linux hosts to act as intelligent routers. This is implemented as modifications to the GateD routine, on machines with multiple Ethernet interfaces. However, we expect that the load of encryption will affect the performance of the router. The Hardened Routers will communicate with the network controllers, located at other Linux hosts. To ensure that the network controllers are hard to compromise (they are on the Internet); we are developing a scaled down version of TCP-IP, which is expected to have fewer (if any) vulnerabilities.

To enhance the performance of the router, we are working on programming the Intel IXP1200 network processor to act as the Hardened Router. This work is in progress and performance data is not available as of yet.

8. Conclusion

In this paper, we discussed the strategies that our "Hardened Network" uses to detect and respond to DDoS attacks. The methods outlined call for embedding cryptographic and control protocols into the fabric of the network, and keeping a consistent, compatible TCP/IP interface to all hosts, as well as to un-upgraded networking ASes. This system can be deployed incrementally. Our simulation shows that the Network can benefit from this infrastructure even in cases where only a small number of the devices are hardened, even though this approach is more effective when more Hardened Routers exist.

Reference:

[1] CERT Coordination Center, "Denial of Service Attacks," http://www.cert.org/tech_tips/denial_of_service.html.

[2] Gary C. Kessler, "Defenses against Distributed Denial-of-services Attack," SANS Reading Room, Threats & Vulnerabilities, November 29th, 2000.

[3] Robert Stone, CenterTrack: An IP Overlay Network for Tracking DoS Floods. In 9th Usenix Security Symposium, August 2000.

[4] John Ioannidis, S.M.Bellovin, "Implementing Pushback: Router-Based Defense Against DDoS Attacks,"

[5] Jelena Mirkovic, Peter Reiher and Gregory Prier, "A Source Router Approach to DDoS Defense," in Proceeding of USENIX Security Symposium 2001.

[6] Deokjo Jeon, "Understanding DDoS Attacks, Tools and Free Anti-tools with Recommendation," SANS Reading Room, Threats & Vulnerabilities, April 7th, 2001.

[7] Ion Stoica and Hui Zhang, "Providing Guaranteed Services without per Flow Management," in SIGCONN'99, 1999, PP. 81-94.

[8] Dawn Xiaodong Song and Adrian Perring, "Advanced and Authenticated Marking Schemes for IP Traceback," in Proceedings of the 2001 IEEE Infocom Conference.

[9] S.M.Bellovin, ICMP Traceback Messages. Internet Draft: Draft-bellovin-itrace-00.txt, Mar. 2000.

[10] Shu Zhang, and Partha Dasgupta, "Hardened Networks: Incremental Upgrading of the Internet for Attack Resilience," ASU CSE Technical Report no. TR-02-006.

[11] Gil, T. M. (2000), "MULTOPS: a Data-structure for Denial-of-service Attack Detection", Master's thesis, Division of Mathematics and Computer Science, Vrije Universiteit, http://citeseer.nj.nec.com/gil00multops.html.