

Using Strategic Firewall Placement to Mitigate the Effects of Distributed Denial of Service Attacks

Wade Chatam
Auburn University



Using Strategic Firewall Placement to Mitigate DDoS



Introduction

- **A Denial of Service (DoS) attack occurs when a malicious user attempts to misuse a system in a way that completely consumes all available resources and, therefore, blocks all services to legitimate users.**
- **A DoS attack typically consumes bandwidth but can also consume memory, CPU cycles, file space, or any other resource that is necessary for normal operation.**
- **Unless countermeasures are taken, the victim will remain at the mercy of the hacker for the entire duration of the attack.**



Denial of Service Overview

- During a typical denial of service attack, an attacker will send massive amounts of bogus traffic to the victim.
- The victim will become overwhelmed by the overload of traffic and will not be able to respond to legitimate users.
- Many times the hacker will use packets with spoofed IP addresses, making it much more difficult to determine which packets are malicious.



Denial of Service Problems

- In February 2000, the online companies Buy.com, ! Yahoo, eBay, Amazon, CNN.com, ETrade, and ZDNet were all taken off the Internet by a distributed denial of service attack.
 - These crimes were committed by a 17- year- old boy with no formal education or training.
 - Special tools to help launch denial of service attacks are available for free on the Internet.
 - The most popular of these tools are Sub7 and BackOrifice.
- Recently, an FBI survey was released that listed DoS as the most monetarily damaging form of electronic attack against companies.
 - Last year, theft of proprietary information was the only type of attack that ranked higher.



Damage from Denial of Service

- If a silver lining must be found in DoS attacks, it arises from the fact that most of the damage incurred from the attacks is due to a loss of revenue.
- After the attack has subsided, the victim's machines can typically be restored with little more than a reboot, and hardware is not usually permanently damaged.
- DoS attacks do not typically give the attacker access to private information or sensitive data.



Distributed DoS

- In order for a hacker to overload a victim, the hacker must be able to produce more traffic than the victim can handle.
 - This is often difficult for a single computer to accomplish.
- To make his/ her attacks more effective, the hacker will accumulate many computers to use in the denial of service attack.
 - This is referred to as Distributed Denial of Service (DDoS).
- A compromised machine that is used in a DDoS attack is typically referred to as a 'zombie' or 'bot.'



Distributed DoS

- **When a hacker wishes to build an army of zombies, he will compromise as many computers as possible.**
 - This can be done using vulnerabilities in the operating system or by tricking the user into running malicious software.
- **When backdoor software is used, the hacker will distribute the program to as many unsuspecting users as possible.**
 - This can be done via email, newsgroups, Web downloads, or any other method of distributing files on the Internet.



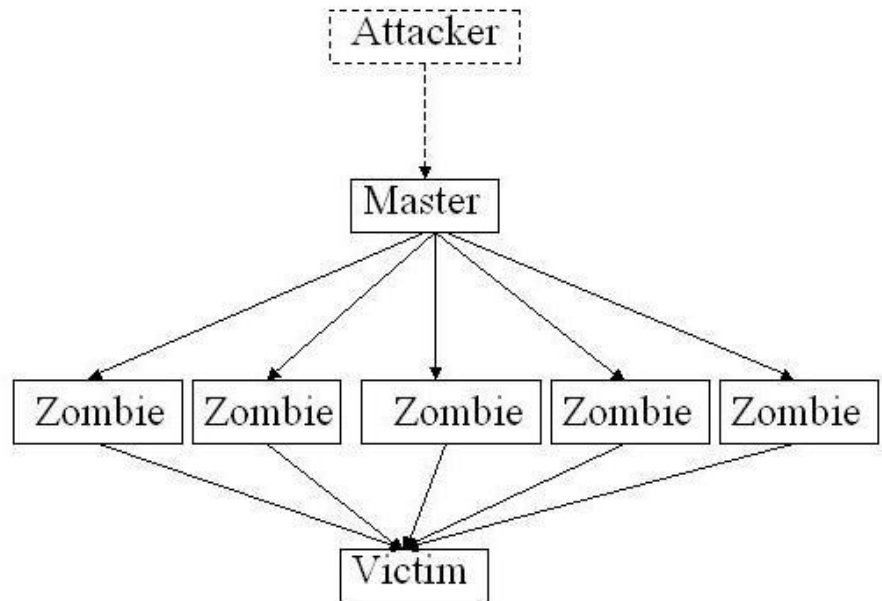
Distributed DoS

- **Once the user has run the backdoor software on his machine, the hacker is given full access to use the computer for any purpose.**
 - The hacker can install a keystroke logger to get credit card numbers, format the hard drive, or launch denial of service attacks.
- **The backdoor software programs available on the Internet have taken extreme care to operate silently without drawing attention from the user.**
 - Most of the time, the user will be completely unaware that his machine has been compromised and is being used for illegal purposes.



Distributed DoS

- When enough zombies have been collected to launch an attack, the hacker will designate one machine as a master.
- The master will then be used to give the attack command to the other zombies.
- This provides one more level of separation between the hacker and the victim.



Industry Basis

- Much of this research has been based on the Gibson Research Corporation (<http://grc.com>)
.
- GRC is a computer security company and has been the victim of multiple distributed denial of service attacks.
 - Unlike Amazon and Yahoo, GRC has posted detailed information concerning the attacks to their web site.
- GRC lists the amount of traffic received at their site during the DDoS attacks, as well as the specific machines and network topology they were using.



GRC

- GRC's ISP, Verio, is connected to the Internet via two 100Mbps trunks, while GRC is connected via two 1.54 Mbps T1 lines.
- During the DoS attacks, more traffic was received than could be handled by the T1 lines.
 - Packets were buffered and eventually dropped at the ISP's router.

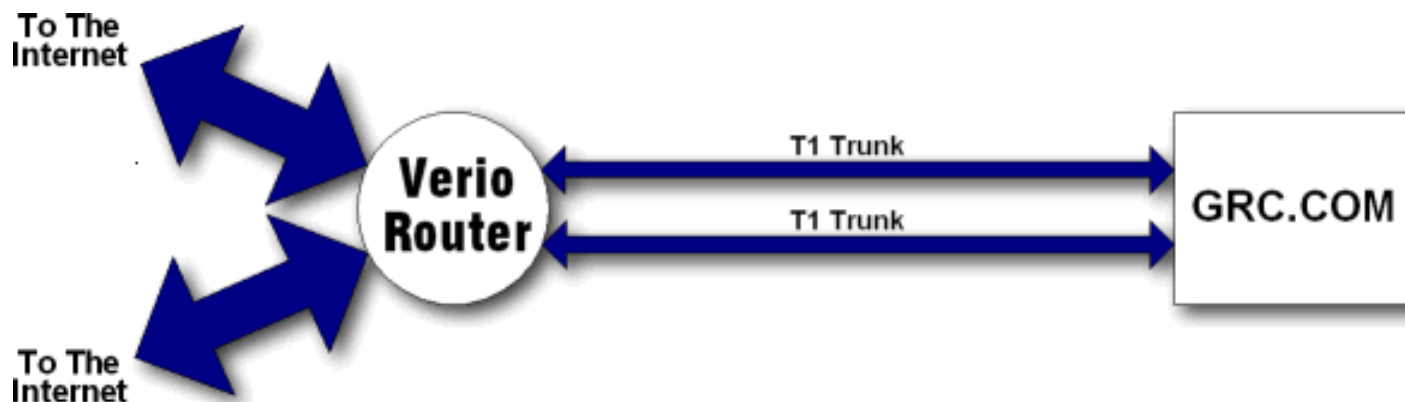
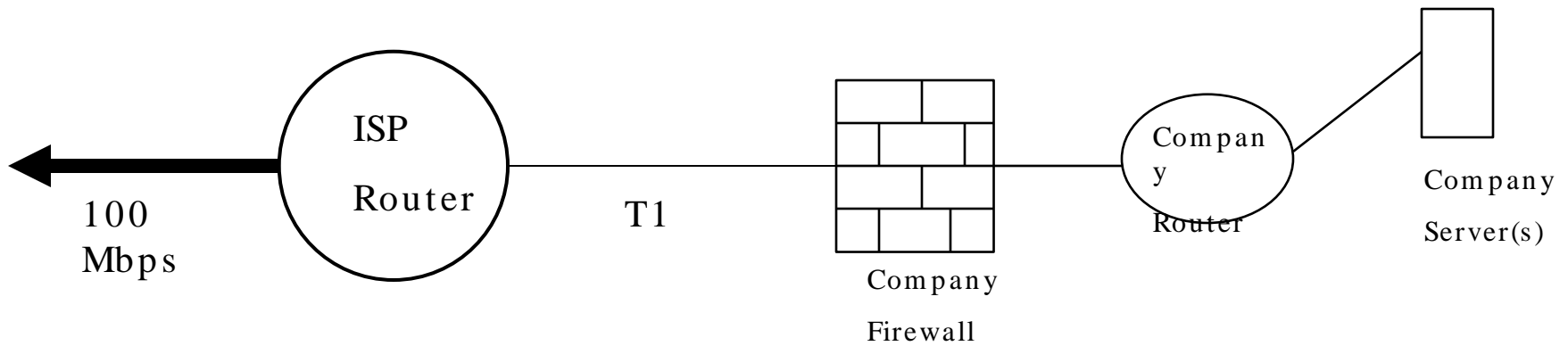


figure 3.



Common Small Business Internet Connection



Using Strategic Firewall Placement to Mitigate DDoS



During an Attack

- **During a bandwidth- flooding attack, packets will arrive at a rate higher than the T1 can handle.**
- **Packets will be dropped at the ISP before they ever reach the victim.**
 - **Regardless of the strength of the victim's security policies, the business is practically helpless when the packets are dropped at the ISP's router.**
 - **ISP is required to filter packets in order to mitigate the attack**

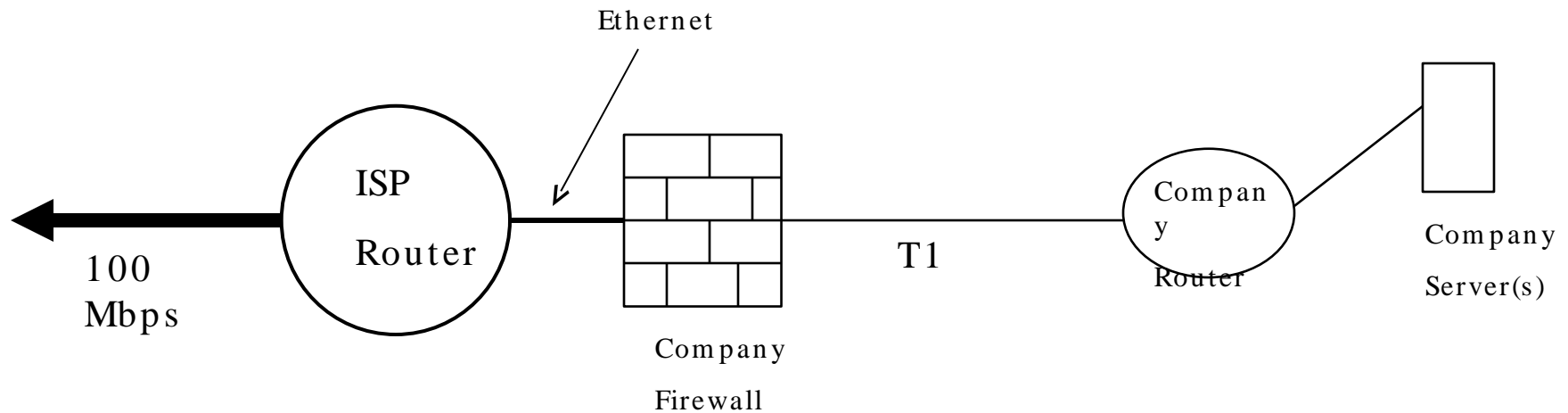


Attack Packets

- **Many of the attacking computers will be sending packets with spoofed IP addresses.**
 - This makes it extremely difficult to determine which packets are malicious based on their IP address.
- **Something must be done in order to filter the packets before they reach the victim's Internet connection link.**



Proposed Network Topology



Using Strategic Firewall Placement to Mitigate DDoS



Firewall Rules

- **Default Deny**
- **When a SYN packet arrives, send SYN/ ACK without forwarding**
 - **Could use Dan Bernstein's TCP SYN cookies**
- **When traffic arrives, check to see if a connection has been established**
- **Monitor outgoing SYN packets in order to know which SYN/ ACK to accept.**



Possible Problems

- This research attempts to answer the question of whether or not this is even feasible.
- Every incoming packet must be compared against a list of known connections that have been established.
- Would it be possible to efficiently check each packet on arrival?
- Can this be done using today's hardware?

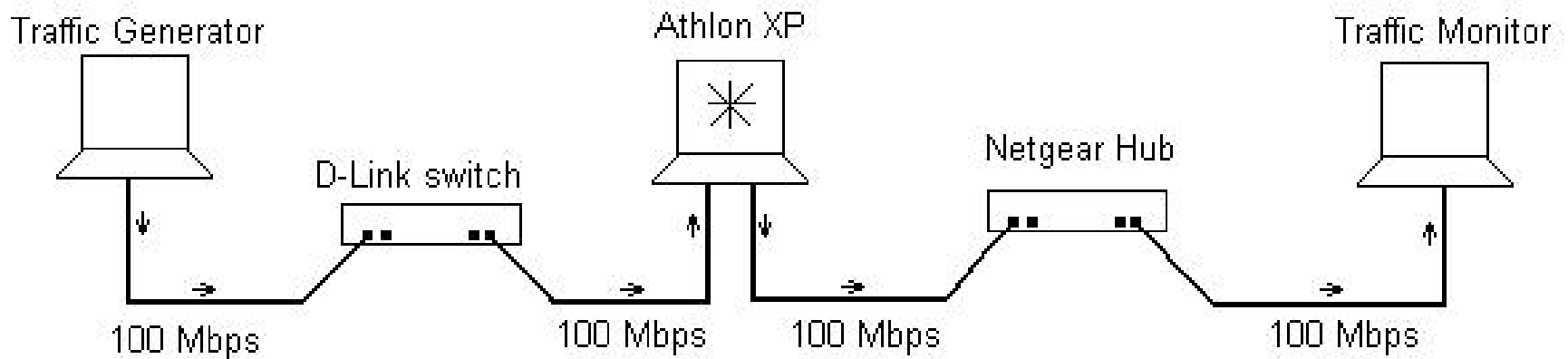


Implementation

- This experimentation is more of a proof-of-concept than a full implementation
- Firewall machine specifics
 - Athlon XP 1500+
 - 256 MB RAM
 - 2 100BaseT Ethernet cards
- Used FreeBSD 5.2.1 as the operating system
- In FreeBSD, each incoming IP packet is processed by the ip_input routine in ip_input.c
 - This file is where all modifications were made



Implementation



Implementation

- A Binary Search Tree was used to store all of the nodes

```
struct ip_treeNode {  
    u_long key;  
    u_long struct ip_treeNode* leftChildPtr;  
    u_long struct ip_treeNode* rightChildPtr;  
    u_char flags;  
};
```

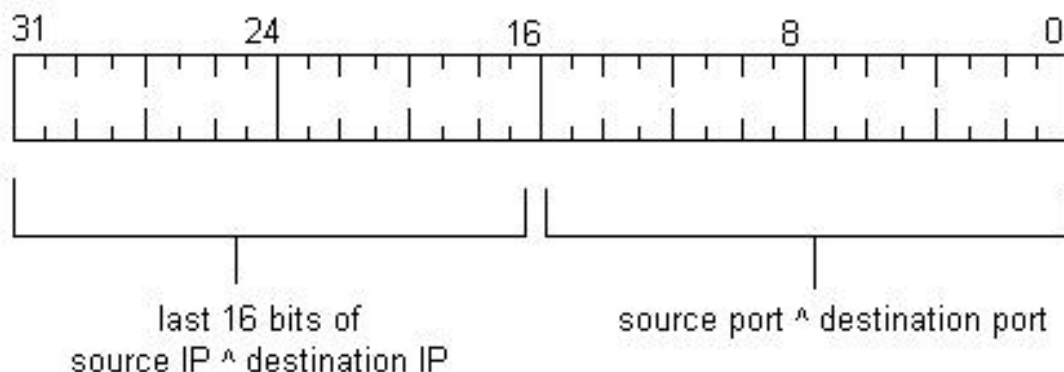
- Each node only consumes 13 bytes of memory
- Binary search trees have average search times of $O(\log N)$.
 - A tree with 1 million nodes would average less than 20 comparisons.
- Tree begins with a static variable `_root`
 - Key value is $2147483647 = 2^{31} - 1$



BST Key

- Key is 32 bits
- First 16 bits are the last 16 bits of the source IP addresses added using one's complement to the last 16 bits of the destination IP address.
- Last 16 bits of key are the one's complement of the source and destination port numbers.

```
u_long key = (ip->ip_dst.s_addr ^ ip->ip_src.s_addr) << 16;  
key += (th->th_dport ^ th->th_sport);
```



Key Advantages

- **Advantages to using this type of key:**
 - Calculation is very fast. It is computed using addition and bitwise shifts.
 - Key is same regardless of source/ destination.
 - For example, if port numbers were not added using one's complement and, instead, were stored separately in the key, incoming and outgoing packets would contain different keys.
 - The source port of the incoming packet would become the destination port in the outgoing packet. Using other techniques, this could produce two different keys.
 - Because the IP addresses and ports are modded, order is unimportant.



Node removal

- Removing nodes from a binary search tree can be time consuming, even if the tree does not have to be balanced.
- Instead, the *flags* variable is used to mark the connection as deleted.
 - If a connection is reestablished, as many often are, the *flags* variable is simply cleared.
- Because nodes are not deleted, the tree will eventually fill up.
 - Instead of attempting to prune the tree, send RST packets to every open connection and then drop the entire tree.



Testing

- **attack_traffic.c**
 - Used raw sockets to send packets with randomly spoofed IP addresses
- **user_traffic.c**
 - Also used raw sockets to send packets from random hosts. The number of hosts sending traffic through firewall could be set. The only difference between the user traffic and the attack traffic is that the users established connections before sending traffic.
- **build_tree.c**
 - Established connections for a given number of nodes in order to build the size of the tree.



Results

Tree Size	Connections	User Rate in Mbps	Attack Rate in Mbps	Total Packets Sent	Packets Dropped	%
100,000	1,000	15.8	0.0	129,500	52	0.04%
100,000	1,000	7.5	22.1	241,322	6	0.00%
100,000	10,000	5.6	21.8	240,932	58	0.02%
500,000	50,000	19.1	0.0	1,144,535	64	0.01%
500,000	50,000	6.5	22.2	1,197,116	7,731	0.65%
1,000,000	100,000	6.6	21.9	146,909	15	0.01%
1,000,000	100,000	7.1	17.2	667,112	936	0.14%
2,000,000	250,000	15.3	36.2	4,275,796	7,252	0.17%
10,000,000	Crashed at approximately 5 million nodes in tree					



Conclusion

- **Based on the results of the tests, this type of firewall implementation is feasible and could be used to mitigate the effects of distributed denial of service attacks.**
- **This type of protection should help businesses who can solicit help from their Internet service providers.**
- **Also useful in government/ military applications where the user has complete control over the network to be protected.**

