Shafiq, Mohammed O. and Ali, Arshad and Ahmed, Ejaz and Ahmed, Hafiz F. and Suguri, Hiroki (2005) ***Detection and prevention of distributed denial of services attacks by collaborative effort of software agents, first prototype implementation***. In: Parallel and Distributed Computing and Networks : Proceedings of the 23rd IASTED International Multi Conference on Applied Informatics, Feburary 2005, Innsbruck, Austria.

Detection and Prevention of Distributed Denial of Services Attacks on Wide Area Networks by Collaborative effort of Software Agents

M. Omair Shafiq, Arshad Ali, Ejaz Ahmad
National University of Sciences and Technology (NUST)
NUST Institute of Information Technology
Chaklala Scheme III, Rawalpindi, Pakistan
Email: {34omair,arshad.ali,ejaz}@niit.edu.pk

H. Farooq Ahmad, Hiroki Suguri
Communication Technologies (Comtec)
2-15-28 Omachi, Aoba-ku, Sendai,
980-0804, Japan
Email: {farooq,suguri}@comtec.co.jp

ABSTRACT
Distributed Denial of Services DDoS, attacks has become one of the biggest threats for resources over Internet. Purpose of these attacks is to make servers deny from providing services to legitimate users. These attacks are also used for occupying media bandwidth. Currently intrusion detection systems can just detect the attacks but cannot prevent / track the location of intruders. Some schemes also prevent the attacks by simply discarding attack packets, which saves victim from attack, but still network bandwidth is wasted. In our opinion, DDoS requires a distributed solution to save wastage of resources. The paper, presents a system that helps us not only in detecting such attacks but also helps in tracing and blocking (to save the bandwidth as well) the multiple intruders using Intelligent Software Agents. The system gives dynamic response and can be integrated with the existing network defense systems without disturbing existing Internet model. We have implemented an agent based networking monitoring system in this regard.

KEYWORDS
Distributed, denial of services, intruders, monitoring, software agents

1. Introduction

Denial of Services (DoS) attack is a method of attacking a server by sending an abnormally high volume of requests on a network, which causes the performance of the servers slowed down. Consequently it becomes unable to provide services to legitimate users. Most common types of DoS attacks [1] are given below:

1.1 Distributed Denial of Services Attacks

Distributed denial-of-service attacks (DDoS) attacks consist of a large number of packets being sent from multiple intruders/attackers to a victim. These packets arrive in such a huge quantity that the resources at the victim like bandwidth, buffers or CPU time is quickly exhausted. The victim either crashes or spends so much time handling the attack that it cannot attend to its real work. Thus legitimate clients are deprived of the victim's service for as long as the attack lasts. These attacks are widely regarded as a major threat to the Internet. They have adversely affected service to individual machines, major Internet commerce sites, and even core Internet infrastructure services.

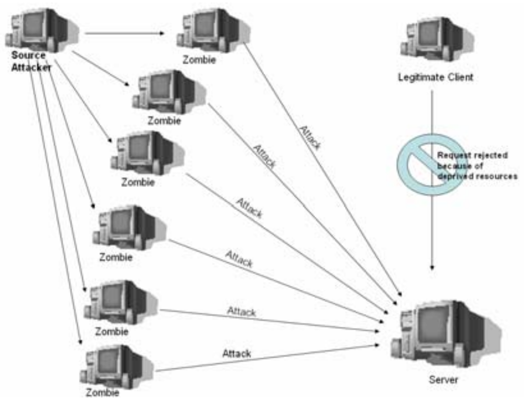

Figure 1. Distributed Denial of Services Attack

Denial-of-service (DoS) and distributed-denial-of-service (DDoS) attacks pose a grave danger to Internet operation. They are, in essence, resource overloading attacks. The goal of the attacker is to tie up a chosen key resource at the victim, usually by sending a high volume of seemingly legitimate traffic requesting some service from the victim.

The over-consumption of the resource leads to degradation or denial of the victim's service to its legitimate clients.

Many of these transactions must be processed in a timely manner and can be seriously delayed by the over utilization of the resources of victim by DDoS attack. The seriousness of the threat is further increased by the ease with which these attacks are performed. Any unsophisticated user can easily locate and download DDoS tools and engage them to perform successful, large-scale attacks. The attacker runs almost no risk of being caught. All of these characteristics have contributed to a widespread incidence of DDoS attacks reports more than 12,000 attacks per week.

There are several features of DDoS attacks that severely challenge the design of TCP model [2] and security defenses:

1.2.1 Use of IP spoofing. Attackers frequently use source address spoofing during the attack they use fake information in the IP source address field in attack packet headers. One benefit attackers receive from IP spoofing is that it is extremely difficult to trace them.

The other advantage that IP spoofing offers to the attackers is the ability to perform reflector attacks. The attacker requests (in the victim's name) a public service that generates large replies to specific small-size requests (amplification effect). The attacker generates as many requests for service as his resources permit, faking the victim's source address, and sends them to public servers.

1.2.2 Large number of attacker machines. Even if trace-back could be successfully performed in the face of IP spoofing, it is difficult to say what actions could be taken against hundreds or thousands of attacker machines. Such a large number prevents any but crude automated responses aimed at stopping attack flows close to the sources.

1.2.3 Similarity of attack to legitimate traffic. Attackers tend to generate legitimate-like packets to perform the attack, obscuring the malicious flow within legitimate traffic [3]. Since malicious packets do not stand out from legitimate ones, it is impossible to sieve legitimate from attack traffic based purely on examination of individual packets. A defense system must keep a volume of statistical data in order to extract transaction semantics from packet flows and thus differentiate some legitimate traffic from the attack traffic.

1.2.4 Attacks to choke network bandwidth. In previous types of attacks, attacker used to spoof source address and destination address remains same because objective is to target one victim. But there are some types of attacks in which objective is, not a single victim but whole network. Such types of attacks are used to choke the network bandwidth. In these types of attacks, spoofing is done not only of source address but also the destination address as given in figure 2 below:

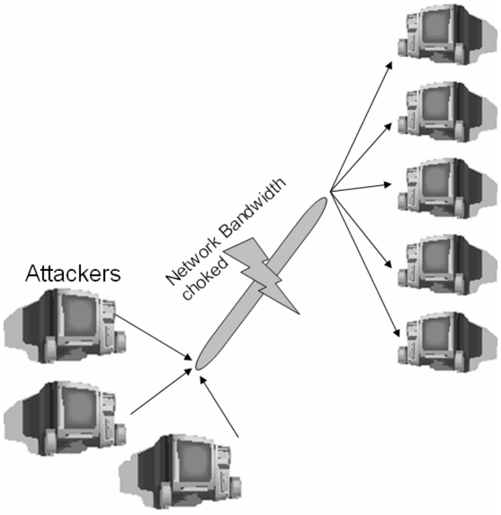

Figure 2. Network bandwidth choked

2. Internet design flaws which makes attacks successful

Design of internet [11], opens security issues and create opportunities for attackers. A few of those are mentioned below:

2.1 Interdependency of Internet security:

Distributed Denial of Services attacks are independent of the security implementation of victims; its susceptibility depends upon the state of security in rest of global internet.

2.2 Internet control is distributed

Internet comprises a lot of networks which run according to local policies defined by the owners. There is no way to define and implement any global internet policy.

2.3 Internet resources are limited

Each internet entity (host, network or service) has limited set of resources that can be consumed by a large number of users, which means that each and every DDoS attempt will be successful in case of absence of defenses.

2.4 Intelligence and resources are not collocated

The end-to-end communication leads to storing most of the intelligence needed for service guarantees with end hosts, limiting the amount of processing in the intermediate network so that packets could be forwarded quickly and at minimal cost. At the same time, a desire for large throughput led to the design of high bandwidth pathways in the intermediate network, while the end networks invested in only as much bandwidth as they thought they might need. Thus, malicious clients can misuse the abundant resources of the unwitting intermediate network for delivery of numerous messages to a less provisioned victim.

2.5 Accountability is not enforced

The source address field in an IP packet is assumed to carry the IP address of the machine that originates the packet. This assumption is not generally validated or enforced at any point on route from the source to the destination. This creates the opportunity for source address spoofing the forging of source address fields in packets Source address spoofing gives attackers a powerful mechanism to escape accountability for their actions, and sometimes even the means to perpetrate attacks.

We summarize all the above discussion about problems which the DDoS creates, are they consume resources victim machine so much so that it can’t serve its legitimate clients. One can’t trace the attackers easily. DDoS attacks are too simple to launch. At the same time, due to the DDoS attacks, a huge amount of bandwidth is also wasted as well. Ideally the DDoS attack should be blocked as close to the source of attack as possible.

3. Existing Intrusion Detection Schemes

We have gone through many Intrusion Detection schemes. Working off all the schemes is impossible to mention here. Instead of it we have discussed the common methodology of all such schemes.

The most commonly used methodology for detection of intrusion based attacks is to capturing and analyzing network packets at end host or at network backbone, maintaining statistical information and comparing the traffic pattern to known attack patterns [20]. If found some pattern matching to some attack pattern, traffic destined for victim is blocked at which saves victim from attack but there is a high probability that legitimate users still denied from services because it is quite difficult to distinguish between normal and attack packets.

All of the schemes can detect intrusion attacks on victim end but there are major limitations in such schemes which are given below:

- Captures data [4] at victim or routers does a lot of calculations and graph based analysis, which is an overhead.
- Can’t trace the intruders.
- Not scalable and hence intrusion detection and prevention policies can’t be applied globally.
- Solutions are proposed to change the currently implemented internet standard which is not a good approach
- Although could save victim from attack but couldn’t save wastage of network bandwidth.

From this, we observed that the schemes are confined to end hosts. The scheme which we have proposed is distributed using software agents. Before we move further, let us explain the Software Agent.

4. Proposed System Architecture

In our opinion, DDoS attacks require a distributed solution. Since distributed solution has a lot of overhead of scalability, maintenance, that’s why we are using Agents which can provides code mobility (Mobile Agents). Hence, Agents [14] are key components of our proposition, which will be monitoring devices, making decisions collectively and changing routing policies over the internet. Before we explain our system let us explain briefly, what agents are:

Agent is a computer system, which is situated in an environment that acts on behalf of its user, in an autonomous way, to achieve its objectives. Agent architecture analyzes agents as independent reactive/proactive entities. Agent architecture conceptualizes agents as being made of perception, action, and reasoning components [15]. The perception component feeds the reasoning component, which governs the agents’ actions, including what to perceive next. Agent system architecture analyzes agents as interacting service provider/consumer entities. Agent infrastructure provides regulations that agents follow to communicate and to understand each other, thereby enabling knowledge sharing. Agent infrastructures mostly deal with the communication among agents based on a communication language using common ontological system. Agents require some platform (environment) to reside on, which we can call as multi-agent system (MAS). In broader sense, it is composed of multiple autonomous components showing the following characteristics:

- Each component has partial capabilities to perform a task
- No global system control; subsystems are autonomous
- Actors, resources and services are decentralized

We propose that network and different network devices should be monitored by Agents. We have divided the monitoring tasks of Agents as follows:

4.1 Agents at stub routers

These agents are static and will be monitoring the end hosts. In case of any malicious activity like DoS attack, they will pass the information to agents monitoring intermediate routers.

4.2 Agents at intermediate routers

These agents are static. Each of this type of agent will be monitoring the traffic at intermediate routers and bandwidth utilization between all directly connected routers to the intermediate router the respective agent.

The agents monitoring routers can change policies of routers in order to make the router block any kind of traffic passing through it.

4.3 Auditing Agents

These are mobile agents [17], which can have any monitoring requirements from network administrators. For example, if it is needed to check the bandwidth available from one router to another, in existing scenario we have install different client-server based applications to check the available bandwidth. In case of our system, the computation methodology of available bandwidth will be provided to mobile agents [21], which will be moved to the routers between which bandwidth is to be monitored. In this way different additions and enhancements could be done very easily.

4.4 Message passing between agents

Under different critical scenarios, Agents will communicate with each other in order to make the decisions collectively.

4.5 System's response for a distributed attack destined for a particular victim

In order to explain this, consider the given scenario: Different organizations having servers are providing services over internet. These organizations are connected to their respective stub routers.

The stub routers are then connected to intermediate routers as given below.

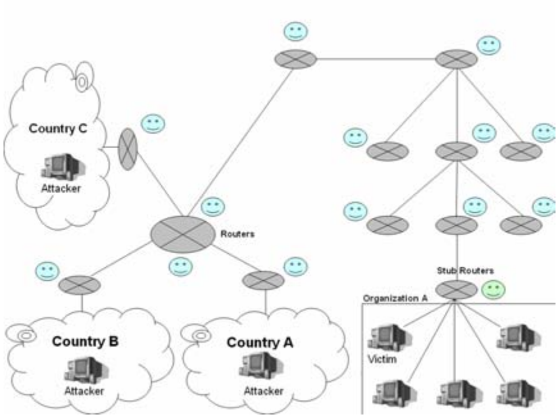

Figure 3. Scenario, Internet topology

Stub routers will be monitored by static agents. These agents will be doing end host monitoring of the servers connected directly to stub router.

Each Agent, monitoring routers like intermediate routers will not have to monitor the servers but the bandwidth of the medium between the respective router and the routers connected directly to it.

Assume that a distributed denial of services attack is launched on a server at some organization by multiple intruders from different places. The agent at stub router will be doing end host monitoring for each host and hence for victim machine as well. As soon as this agent detects the resources of the victim machine to be consumed up to a certain threshold, it will change the policy of stub router and will make the stub router will start dropping some calculated amount of packet destined for victim (so the resources at machine couldn't be over consumed and attack couldn't crash machine) until it inform all the agents at monitoring at routers which are directly connected with the stub router.

Agent at stub router has detected the attack and has informed agents at intermediate routers and hence given responsibility to agents of intermediate routers. After doing this, the agent at stub router will change the policy of its router to normal.

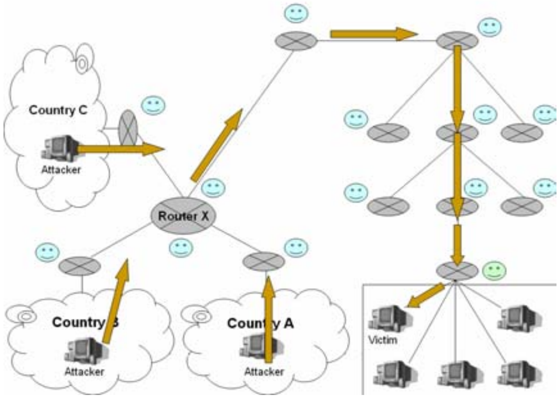

Figure 4. Distributed intrusion based attack

Now agents at intermediate routers will check for the same type of attack on each of its interface and the attack detected at some interface, information will be passed to the routers connected at that interface in the same way as previously described until it is reached to intruders. When the intruders will be identified, then it will become very easy to identify the attack traffic. Moreover, when attack traffic will be blocked on intruder's side, it will save a lot of network bandwidth as well.

4.6 System's response for a distributed attack to choke network bandwidth

When multiple intruders start throwing packets on network for choking network bandwidth, in this case packets are not destined for a particular victim. This means, destination IP address will also be spoofed in this case.

As we have mentioned earlier that static agents at intermediate routers will be monitoring network bandwidth of medium between other directly connected routers. Suppose some attack is launched with objective to choke the network bandwidth of a particular medium between two routers. The agents monitoring the routers will ensure that the bandwidth could not be consumed more than a certain threshold. As soon as bandwidth consumption reaches the certain threshold, direction of incoming traffic will be determined by the agents of the respective routers. The agent at router from which traffic is incoming will communicate the attack information of attack to the agents at neighbor routers and a calculated amount of packets (to avoid over consumption of bandwidth) from router attached each interface will be dropped to minimize the probability of discarding legitimate traffic which can be called as false positive.

5. First Prototype Implementation: Agent based networking monitoring system

In the proposed system we need different types of monitoring agents doing different tasks to be developed:

5.1 End host monitoring application

This application will be used to monitor the information of end hosts like CPU utilization, physical memory utilization, kernel memory utilization, processes, threads, running, traffic capturing, different TCP and ICMP statistics.

5.2 Agents monitoring traffic at stub routers

These agents will be monitoring hosts connected directly using the above application.

5.3 Agents monitoring traffic at intermediate routers

These agents will be monitoring traffic at their respective routers and the bandwidth by different bandwidth utilization algorithms.

5.4 Agents changing policies of routers

All the agents on routers will not be residing on the routers but capturing traffic and changing the policies of routers accordingly using terminal services (controlling routers). This means, agents will control all the system and there will be no need of human interaction.

5.5 Communication between agents

It will help them work collectively. Since while system is being controlled by agents and different tasks are assigned to different agents. Any agent after detecting some anomaly will inform other agents by passing ACL (Agent communication Language) message.

5.6 Mobile Agents

Mobile agents [17] are also very important part of this system. Our system is completely distributed. Mobile agents will help in maintenance and changing in working of different schemes. For examples the agents at different remote routers are calculating bandwidth among each other by some algorithm and it is required to measure bandwidth between the remote routers by using another efficient algorithm. This can be done by using mobile agents. Instead of changing the implementation at routers where bandwidth is needed to be measured, Mobile agents having the new algorithm could be sent at the routers and can measure the bandwidth using the new algorithms and can provide results.

6. Conclusion

Most commonly used intrusion detection schemes for distributed attacks can't accurately distinguish between legitimate and attack traffic at victim end and can't trace the intruders as well. Distributed attacks detection and prevention systems require a distributed solution. At the same time distributed solutions are not easy to implement and maintain. Keeping in mind all such things, we have used software agents which provide code mobility that will help in maintenance of the distributed solutions like updating. Other monitoring agents are provided to change in policy implementation at different situations. Agents work collectively and make decisions at their own which make the scheme maintain network with out any human interruption. These agents help in tracing intruders as well. Since agents work on application layer and hence need not any change in existing internet standards and it will not cause burden on network devices as Agents will run and monitor traffic using terminal services of routers. It is scalable as well because if some of the intermediate routers doesn't support the agents, will be ignored by the system.

7. References

[1] Intrusion detection systems: Protocol Anomaly Detection White Paper – Symantec Enterprise Security.

[2] Security problems in TCP/IP protocol suite
By: S. M. Bellovin, Computer Communication Review, April 1989.

[3] CERT/CC Security Statistics during 1998 2002, http://www.cert.org/stats/cert_stats.html, Oct 17, 2003.

[4] Detection of Denial of Service Attacks using AGURI
By: Ryo Kaizaki, Kenjoro Cho, Osamu Nakamura.

[5] Denial of Service: By: R. Needham, Communications of the ACM vol. 37, Nov 1994.

[6] Smurf IP D-o-S Attacks http://www.cert.org/advisories/CA-1998-01.html, March 13, 2000.

[7] TCP SYN Flooding & IP Spoofing Attacks http://www.cert.org/advisories/CA-1996-21.html, Nov 29, 2000.

[8] D-o-S Attack via ping http://www.cert.org/advisories/CA-1996-26.html, Dec 5, 1997.

[9] Trends in D-o-S Attack Technology, By: Kevin J. Houle, George M. Weaver
http://www.cert.org/archive/pdf/DoS_trends.pdf, Oct 2001

[10] Spider Magazine:Cyber War (Aug '02 issue) By Sameer Qureshi http://spider.tm/aug2002/covcyber.shtml, August 2002.

[11] TCP/IP Illustrated Vol I By: W. Richard Stevens.

[12] Resisting SYN Flooding DoS Attacks with a SYN Cache By: J. Lemon Proceedings of USENIX BSDCon' 2002, Feb'02.

[13] M. Milenkovic, Scot H. Robinson, Rob C. Knauerhase, D. Barkai, S. Garg, V. Tewari, Todd A. Aderson, M. Bowman, "Toward Internet Distributed Computing", IEEE Computer, vol.36, no.5, pp.38-46, (2003).

[14] N. R. Jennings, K. Sycara, and M. Wooldridge. A
roadmap of agent research and development. Int Journal of Autonomous Agents and Multi-Agent Systems, 1(1):7–38, 1998.

[15] FIPA: Foundation for Intelligent Physical Agents, see http://www.fipa.org, and P.D. O'Brien and R. Nicol, "FIPA: Towards a standard for intelligent agents." BT Technical Journal, 16(3), 1998.

[16] Mobile agent applications, By: Dejan Milojicic, Hewlett Packard Laboratories,
http://www.computer.org/concurrency/pd1999/pdf/p3080.pdf.

[17] Active Monitoring in GRID environments using Mobile Agent technology By: Orazio Tomarchio, Andrea Calvagna at Dipartimento di Ingegneria Informatica e delle Telecomunicazion ,Università di Catania (ITALY).

[18] Asaka, M., Okazawa, S., Taguchi, A., Goto, S. A method of Tracing Intruders, INET99, June 1999.

[19] Port Scanning, A popular reconnaissance technique.
(http://www.networkice.com/advice/underground/hacking/methods/technical/port%5Fscan/flags/default.htm).

[20] Snort - The Open Source Network Intrusion Detection System. (http://www.snort.org).