International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 3 Issue 2 May 2012

Study of DDoS attacks using DETER Testbed

Daljeet Kaur
Asst. Proff.
Deptt.of Computer Sc.& Engg
SBS College of Engg.& Tech-India
91-8054100727
daljeetkaur617@gmail.com

Monika Sachdeva
Asst. Proff.
Deptt. of Computer Sc. & Engg.
SBS College of Engg & Tech-India
91-9463097771
monika.sal@rediffmail.com

Krishan Kumar
Associate. Proff.
Deptt. of Computer Sc. & Engg.
Punjab Institute of Technology- India
91-8054100707
k.saluja@rediffmail.com

ABSTRACT

In present era, the world is highly dependent on the Internet and it is considered as main infrastructure of the global information society. Therefore, the Availability of information and services is very critical for the socio-economic growth of the society. However, the inherent vulnerabilities of the Internet architecture provide opportunities for a lot of attacks on its infrastructure and services. Distributed denial-of-service (DDoS) attack is one such kind of attack, which poses an immense threat to the availability of the Internet. These attacks not only congest a Server by their attack, but also affect the performance of other Servers on the entire network also, which are connected to Backbone Link directly or indirectly. To analyze the effect of DDoS attack on FTP services, repeated research in cyber security that is vital to the scientific advancement of the field is required. To meet this requirement, the cyber-DEfense Technology Experimental Research (DETER) testbed has been developed. In this paper, we have created dumb-bell topology and generated background traffic as FTP traffic. Different types of DDoS attacks are also launched along with FTP traffic by using attack tools available in DETER testbed. Throughput of FTP server is analyzed with and without DDoS attacks.

Keywords

DDoS, availability, vulnerability, confidentiality, botnet.

1. INTRODUCTION

The “availability” means that the information, the computing systems, and the security controls are all accessible and operable in committed state at some random point of time (Tipton et al., 2004). Threat to the Internet availability is a big issue which is hampering growth and survival of E-business and other Internet based applications. Internet failures can be accidental or intentional. The Internet design concentrates mainly on providing functionality though a little attention has been given on designing strategies for controlling accidental failures. On the other hand, intentional attacks by malicious users have no answer in the original Internet design. A denial-of-service (DoS) is such an intentional attempt by malicious users / attackers to completely disrupt or degrade (compromise) availability of service/resource to legitimate/authorized users (Criscuolo, 2000).

Some well-known DoS attacks are SYN Flood, Teardrop, Smurf, Ping of Death, Land, Finger Bomb, Black Holes, Octopus, Snork, ARP Cache Poisoning and the Misdirection. DoS attacks exploit weaknesses in Internet protocols, applications, operating systems, and protocol implementation in operating systems.

Distributed denial-of-service attacks (DDoS) degrade or completely disrupt services to legitimate users by expending communication and/or computational resources of the target. (Mirkovic et al.,2004) and (Chen et al.,2007) described DDoS attacks as amplified form of DoS attacks, where attackers direct hundreds or even thousands of compromised hosts called zombies against a single target. There are varieties of DDoS attacks as classified in (Mirkovic et al, 2004) (Douligeris et al., 2004). However, the most common form of DDoS attacks is a packet-flooding attack, in which a large number of seemingly legitimate TCP, User Datagram Protocol (UDP), or Internet Control Message Protocol (ICMP) packets are directed to a specific destination.

As per (Moore et al., 2006) defending against these attacks is challenging for mainly two reasons. First, the number of zombies involved in a DDoS attack is very large and deployment of these zombies spans large geographical areas. The volume of traffic sent by a single zombie might be small, but the volume of aggregated traffic arriving at the victim host is overwhelming. Second, zombies usually spoof their IP addresses under the control of attacker, which makes it very difficult to trace the attack traffic back even to zombies. According to the Internet architecture working group (Handley, 2005), the percentage of spoofed attacks is declining, but the sheer volume and distributed nature of DDoS attack traffic still the design of an effective defense.

2. DDOS ATTACKS

An attacker or hacker gradually implants attack programs on these insecure machines. Depending upon sophistication in logic of implanted programs these compromised machines are called Handlers or Zombies and are collectively called bots and the attack network is called botnet in hacker's community. Hackers send control instructions to masters, which in turn communicate it to zombies for launching attack. The zombie machines under control of masters/handlers (running control mechanism) as shown in Figure 1 transmit attack packets, which converge at victim or its network to exhaust either its communication or computational resources.

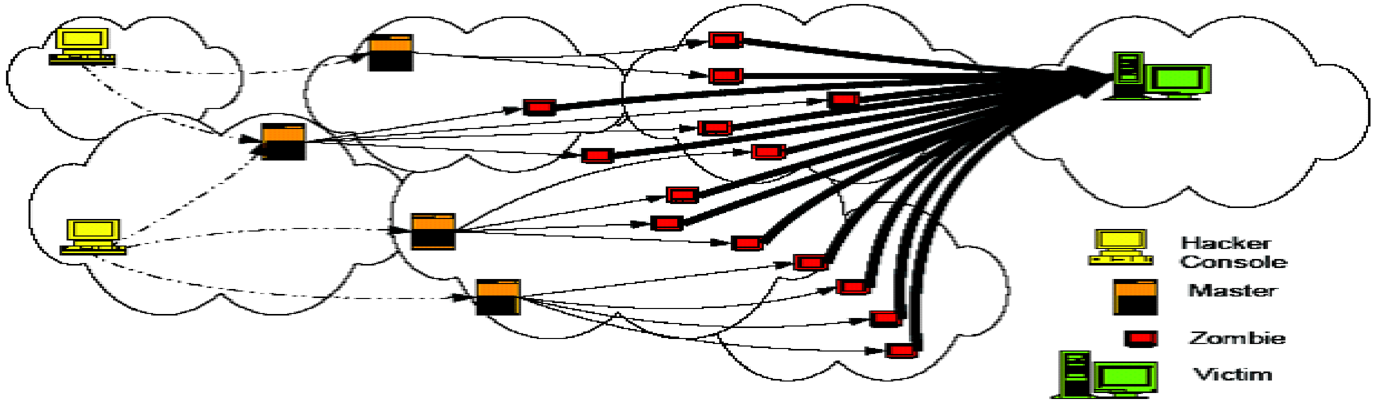

Figure 1 Attack Modus Operandi

(Mirkovic et al., 2004) have classified DDOS attacks into two broad categories: flooding attacks and vulnerability attacks. Flooding DDoS attacks consume resources such as network bandwidth by overwhelming Backbone link with a high volume of packets. Vulnerability attacks use the expected behavior of protocols such as TCP and HTTP to the attacker's advantage. Flooding DDoS is basically a resource overloading problem. The resource can be bandwidth, memory, CPU cycles, file descriptors and buffers etc. A flood of packets congests the link between ISP's edge router and border router of victim domain (Handley, 2005).

The congestion and flow control signals force legitimate clients to decrease their rate of sending requests, whereas attack packets keep coming. Finally, a stage comes when only attack traffic is reaching at the server. Thus, service is denied to legitimate clients. Moreover (Robinson et al., 2003) stated that as attack strength grows by using multiple sources, the computational requirements of even filtering traffic of malicious flows become a burden at the target.

3. RELATED WORK

Theory (Mirkovic et al, 2009) is well-suited to answering questions about situations that can be accurately represented by existing models, such as M/M/1 queues, state diagrams, probabilistic models, hash tables, random selection from a set, etc. In general, theory is a poor choice for effectiveness evaluation. While it may be able to answer sub-questions related to effectiveness, we lack theoretical tools powerful enough to model the complexity of traffic mixes, their dynamics and their interaction with the underlying hardware and network protocols, especially in high-stress situations like DoS. Simulation is highly popular for addressing network performance questions. Network simulators must balance a tradeoff between fidelity and scalability (Nicol , 2003-2). At one end of the spectrum, simulators can choose to sacrifice fidelity, especially at the lower layers of the protocol stack, for scalability. Emulation involves testing in a mini-network, such as a lab or a shared testbed. Three testbeds have been popular for DoS defense testing: Emulab (White et al., 2002), DETER (Benzel et al., 2006) and Planetlab (Peterson et al., 2006). Emulab and DETER allow users to gain exclusive access to a desired number of PCs, located at a central facility and isolated from the Internet. These can be loaded with a user-specified OS and applications, and users obtain root privileges. Emulation offers a more realistic evaluation environment than theory and simulation, for following reasons: 1.A real OS and applications, and real hardware are used in testing. 2. Live legitimate and DoS traffic can be generated and customized in various ways. 3. Several router choices exist in testbeds such as PC, Click, Cisco and Juniper routers, allowing realistic forwarding behavior. Emulation also means testing with a defense's prototype, instead of abstracting the defense and simulating it or developing its theoretical model. This produces higher-fidelity results.

4. DETER TESTBED

The DETER testbed (Benzel et al., 2006) aims to facilitate network security experimentation by providing an environment for researchers to perform experiments within, in a secure, isolated fashion. DETER runs a tailored configuration of the Emulab software developed at Utah (White et al., 2002).

Various network security threats plague today's communication and undermine the Internet's stability and reliability. The DETER testbed (Benzel et al., 2006) was funded by the Department of Homeland Security and the National Science Foundation, and developed by USC Information Sciences Institute and UC Berkeley, with the goal of providing an infrastructure for safe, repeatable and versatile security experimentation. DETER allows security researchers to replicate threats of interest in a secure environment and to develop, deploy and evaluate potential solutions. The testbed has a variety of hardware devices and supports many popular operating systems. Researchers obtain exclusive use of a portion of a testbed, configured into a user-specified topology, and shielded from the outside world via a firewall. DETER's hardware infrastructure was enhanced by a collection of software tools for traffic generation, statistics collection, analysis and visualization, developed in its sister project EMIST (See documentation of EMIST project overview). Jointly, DETER and EMIST facilitates reconstruction of numerous security scenarios, where every element of the scenario is customizable by the researcher.

4.1 Evaluation in Testbed Experiments

We evaluated legitimate traffic with experiments on the DETER testbed using SEER GUI BETA6 environment (Benzel et al., 2006) (Mirkovic et al, 2007). The test bed is located at the USC Information Sciences Institute and UC Berkeley, and allows security researchers to evaluate attacks and defenses in a controlled environment.

4.1.1 Experimental Topology

Figure 2 shows our experimental topology definition and Figure 3 shows the experimental topology, where legitimate client node A1 and attack client node A2 is connected with server node V.

```
set ns [new Simulator]

source tb_compat.tcl

#Create the topology nodes

foreach node { V R A1 A2 control } {

 #Create new node

  set $node [$ns node]

#Define the OS image

 tb-set-node-os [set $node] FC4-STD

 #Have SEER install itself and startup when the node is ready

  tb-set-node-startcmd [set $node] "sudo python /share/seer/v160/experiment-setup.py Basic"
```

Figure 2 Experiment Topology Defination.

4.1.2 Legitimate Traffic

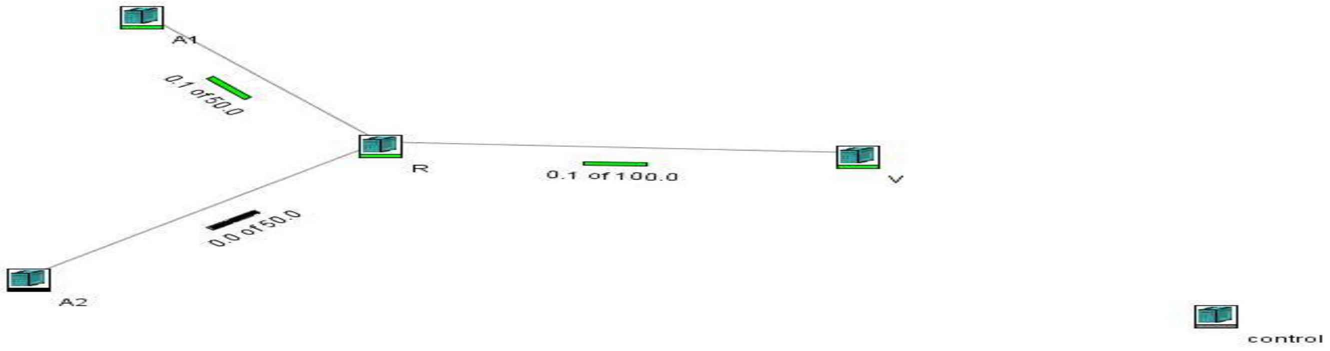

Figure 3. Experiment Topology.

Initially, the legitimate client A1 sends the request to the server V, the request is routed via intermediate node R to the intended server. The server services the request, and replies to clients with their requested file. Thus V, carries the legitimate traffic only as described by Table 1. The overall traffic carried by V is shown in Figure 4.

Table 1. Configuration of legitimate FTP traffic

Client	***Server***	***Thinking Time***	***File Sizes***
A1	V	Minmax(0.01,0.1)	Minmax(512,1024)

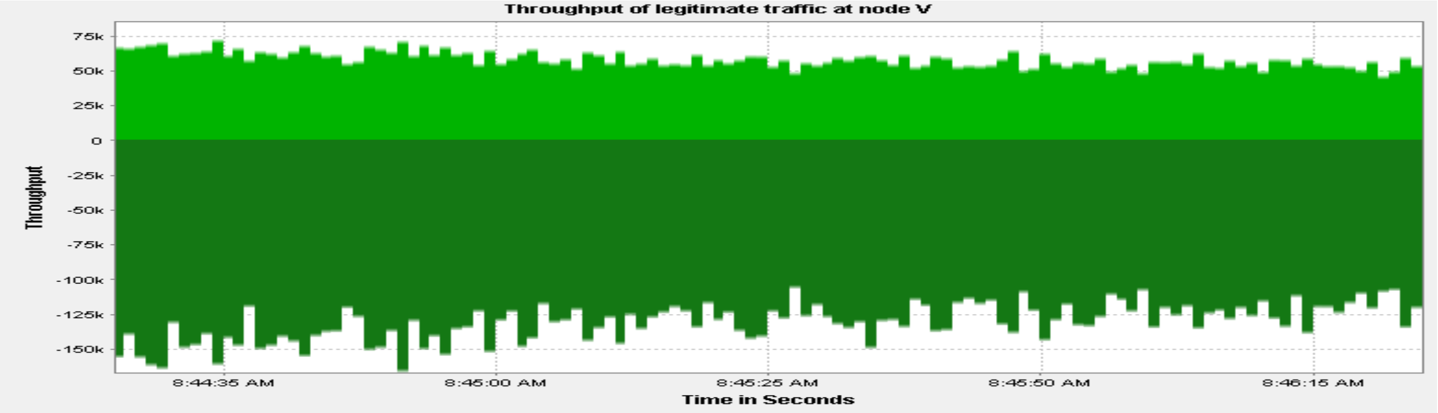

Figure 4. Throughput of legitimate traffic at node V.

4.1.3 Attack Traffic

DDoS packet flooding attack is launched by attacker clients A2 to the victim server V. In this experiment, we have generated UDP and TCP flood with FLAT, PULSE and RAMP distributions to achieve attacks in different scenarios. The attack traffic affects the throughput of server V.

5. RESULTS AND DISCUSSIONS

The attack traffic is routed via node R to the intended destination V. Thus at this point of time the node V contains legitimate traffic requested by node A1 and attack traffic launched by node A2. We have created following emulation scenarios and the performance of FTP server in terms of Throughput is analyzed for all scenarios.

5.1 Using TCP attacks

The configuration of all the Scenarios using TCP attacks is shown in Table 2.

Table2. Configuration of TCP attacks

Attack Type	*Flooding*	*Flooding*	*Flooding*	*Flooding*	*Flooding*
Attack Source	A2	A2	A2	A2	A2
Attack Target	V	V	V	V	V
Protocol	TCP	TCP	TCP	TCP	TCP
Length Min	1	1	1	1	1
Length Max	1	1	1	1	1
Flood Type	Flat	rampup	Ramp down	Pulse	Ramp pulse
High Rate	300	350	500	250	500
High Time	0	5000	5000	5000	5000
Low Rate	0	100	350	100	350
Low Time	0	8000	10000	8000	10000
Rise Shape	0	1.0	0	0	1.0
Rise Time	0	10000	6000	0	6000
Fall Shape	0	0	1.0	0	1.0
Fall Time	0	0	6000	0	6000
Sport Min	57	57	57	57	57
Sport Max	57	57	57	57	57
Dport Min	1000	1000	1000	1000	1000
Dport	2000	2000	2000	2000	2000

Max TCPFlags	SYN	SYN	SYN	SYN	SYN

Throughput at node V at random point of time during TCP attack is shown in Figures 5 - 9.

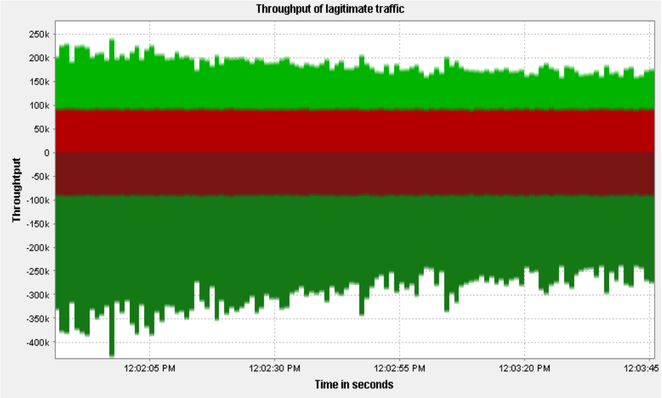

Figure 5.Throughput during TCP flat attack.

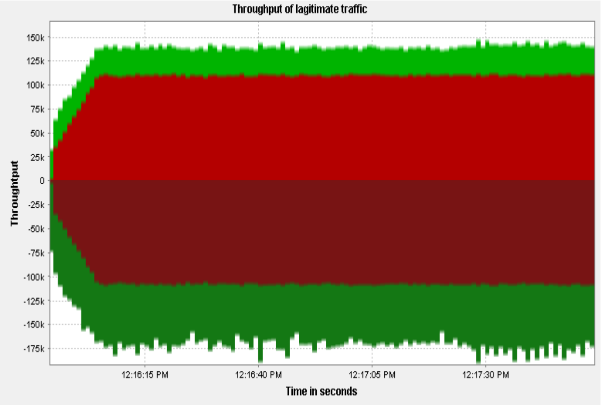

Figure 6.Throughput during TCP rampup attack.

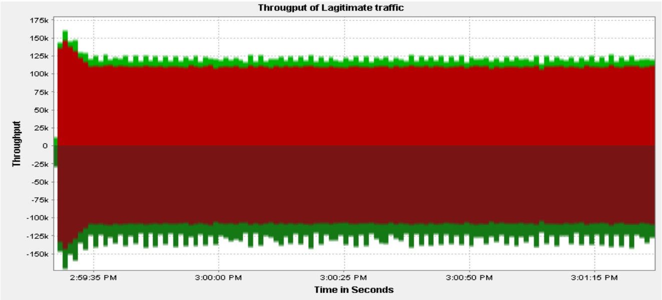

Figure 7.Throughput during TCP rampdown attack.

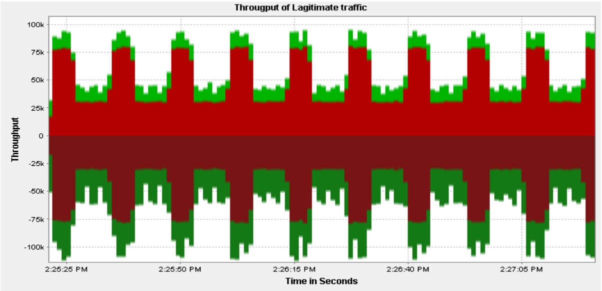

Figure 8.Throughput during TCP pulse attack.

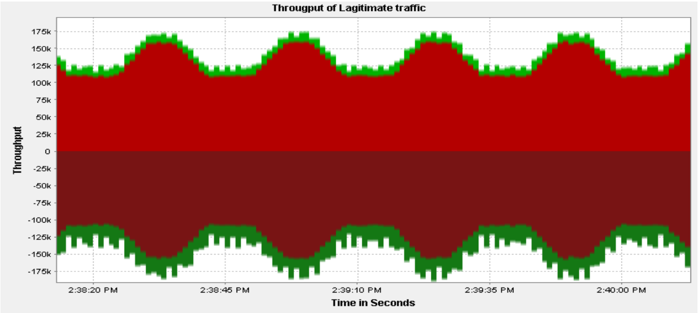

Figure 9.Throughput during ppulse attack. TCP ram

5.2 Using UDP attacks

Attack can also be launched using UDP protocol. The detailed configuration of flat, rampup, rampdown, pulse, ramppulse is demonstrated in Table 3.

- 🟩 Legitimate traffic
- 🟥 Attack traffic

Table3. Configuration of UDP attacks

Attack Type	*Flooding*	*Flooding*	*Flooding*	*Flooding*	*Flooding*
Attack Source	A2	A2	A2	A2	A2
Attack Target	V	V	V	V	V
Protocol	UDP	UDP	UDP	UDP	UDP
Length Min	1	1	1	1	1
Length Max	1	1	1	1	1
Flood Type	Flat	Ramp up	Ramp down	Pulse	Ramp pulse
High Rate	300	350	500	250	500
High Time	0	5000	5000	5000	5000
Low Rate	0	100	350	100	350
Low Time	0	8000	10000	8000	10000
Rise Shape	0	1.0	0	0	1.0
Rise Time	0	10000	6000	0	6000
Fall Shape	0	0	1.0	0	1.0

Fall Time	0	0	6000	0	6000
Sport Min	57	57	57	57	57
Sport Max	57	57	57	57	57
Dport Min	1000	1000	1000	1000	1000
Dport Max	2000	2000	2000	2000	2000

Throughput at node V at random point of time during UDP attack is shown in Figures 10 – 14.

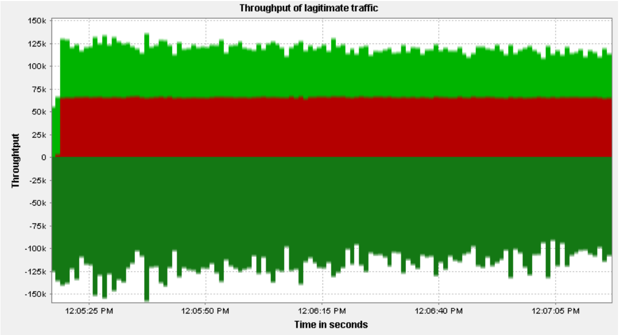

Figure 10 Throughput during UDP Flat attack.

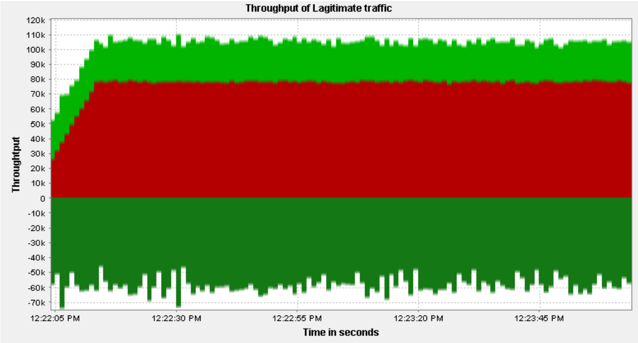

Figure 11 Throughput during UDP rampup attack.

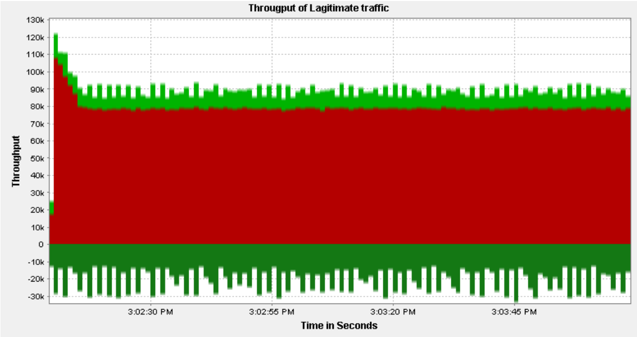

Figure 12 Throughput during UDP rampdown attack.

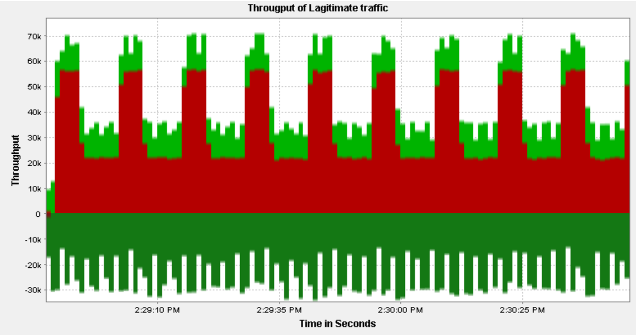

Figure 13 Throughput during UDP pulse attack.

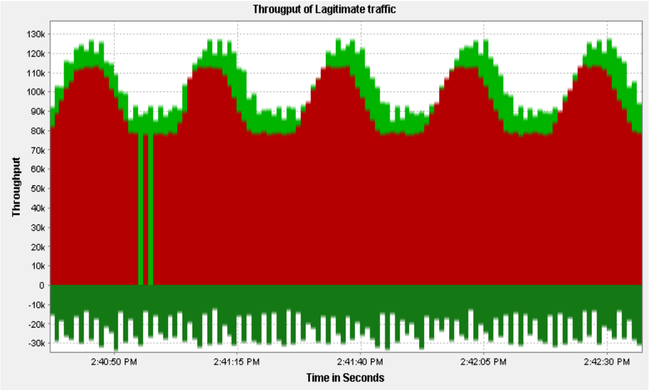

Figure 14. Throughput during UDP ramppulse attack.

6. CONCLUSIONS AND FUTURE WORK

There is alarming increase in the number of DDoS attack incidents. Not only, DDoS incidents are growing day by day but the technique to attack, botnet size, and attack traffic are also attaining new heights. Effective mechanisms are needed to elicit the information of attack to develop the potential defense mechanism. DETER testbed allows to carry the DDoS attack experiment in a secure environment. It also allows creating, plan, and iterating through a large range of experimental scenarios with a relative ease. We pointed out the possibility of DDoS attacks on FTP application by analyzing the characteristics of FTP application. DDoS attacks are launched on FTP server and analyzed throughput of legitimate traffic by using different protocols by Emulating attack scenarios. The future work is to carry, plan and iterate through various range of experimental scenarios and then measure the impact of DDoS attack on internet traffic using some metrics i e Throughput, response time, no of request dropout.

7. ACKNOWLEDGMENTS

We would like to express our gratitude to all those who gave us the possibility to complete this experimental work. We are extremely thankful to all the colleagues and faculty members for their constructive criticism and guidelines.

8. REFERENCES

[1] Benzel, T.; Braden, R.; Kim, D.; Neuman, C.; Joseph, A.; Sklower, K.; Ostrenga, R.; Schwab, S.; Experience with DETER: A Testbed for Security Research. In Proceedings of the 2nd IEEE Conference on Testbeds and research Infrastructures for the Development of Networks and Communities (TridentCom 2006), Barcelona, SPAIN, March 2006.

[2] Criscuolo, P.J, “Distributed Denial of Service Trin00, Tribe Flood Network, Tribe Flood Network 2000, and Stacheldraht CIAC-2319”, Department of Energy computer Incident Advisory (CIAC), UCRL-ID-136939, Rev. 1, Lawrence Livermore National Laboratory, February 14, 2000. http://ftp.se.kde.org/pub/security/csir/ciac/ciacdocs/ciac2319.txt.

[3] Chen, R., Park, J., and Marchany, R., “A Divide-and-Conquer Strategy for Thwarting Distributed Denial-of-Service Attacks,” IEEE Transactions on Parallel and Distributed Systems, Vol. 18, No. 5, pp. 577-588, May 2007.

[4] Douligeris, C. and Mitrokotsa, A., “DDoS attacks and defense mechanisms: classification and state-of-the-art,” Computer Networks, Vol. 44, No. 5, pp. 643–666, April 2004.

[5] EMIST project. Evaluation methods for internet security technology. http://www.isi.edu/deter/emist.temp.html

[6] Handley, M., Internet Architecture WG: DoS-resistant Internet subgroup report, 2005. http://www.communications.net/object/download/1543/doc/mjhdos- summary.pdf..

[7] Mirkovic, J., Fahmy, S., Reiher, P., Thomas, R., “How to test DoS Defence”,2009.

[8] Mirkovic, J., and Reiher, P., “A Taxonomy of DDoS Attack and DDoS Defense Mechanisms,” ACM SIGCOMM Computer Communications Review, Volume 34, No. 2, pp. 39-53, April, 2004.

[9] Moore, D., Shannon, C., Brown, D. J. , Voelker, G., and Savage, S., “Inferring Internet Denial-of-Service Activity,” ACM Transactions on Computer Systems, Vol. 24, No. 2, pp. 115–139, May 2006

[10] Mirkovic, J., Wei, S., Hussain, A., Wilson, B., Thomas, R., Schwab, S., Fahmy, S., Chertov, R., and Reiher, P. “DDoS Benchmarks and Experimenter’s Workbench for the DETER Testbed”, Proceedings of Tridentcom, 2007.

[11] Nicol, D.M. Scalability of network simulators revisited. In Proceedings of the Communications Networking and Distributed Systems Modeling and Simulation Conference, February 2003.

[12] Nicol, D.M. Utility analysis of network simulators. International journal of simulation: Systems, Science, and Technology, 2003.

[13] Peterson, L., Bavier, A., Fiuczynski, M.E. and Muir, S., “Experiences building planetlab.” In Proceedings of the 7th USENIX Symposium on Operating System Design and Implementation (OSDI ’06), Seattle, WA, November 2006.

[14] Robinson, M., Mirkovic, J., Schnaider, M., Michel, S., and Reiher, P., “Challenges and principles of DDoS defense,” ACM SIGCOMM, 2003.

[15] Tipton, H. F. and Krause, M., “Information Security Management Handbook”, CRC Press, 2004

[16] White, B., Lepreau,J., Stoller, L.,Ricci, R., Guruprasad, S., Newbold, M., Hibler, M., Barb, C., and Joglekar, A. An Integratede Experimental Environment for Distributed Systems and Networks. In Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (ODSI), pp. 255-270, Boston, MA, Dec. 2002.