Kolaborasi Sistem Deteksi Berbasis *Non Agent* Untuk Mengatasi Serangan DDoS

Collaboration of Detection System Using Non Agent-Based to Handle DDoS Attack

Ryan Danny Kresnawan[1], Yudha Purwanto[2], Doan Perdana[3]
[1,2,3]Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Telkom University
Bandung, Indonesia
[1]ryandanny@students.telkomuniversity.ac.id, [2]omyudha@telkomuniversity.ac.id , [3]doanperdana@telkomuniversity.ac.id

Abstrak

Perkembangan teknologi yang sangat dinamis, dapat memungkinkan adanya serangan yang dapat mengganggu sebuah komputer atau server yang terhubung dalam suatu jaringan. Salah satu serangan yang dapat terjadi yaitu serangan DoS (*Denial of Service)*. Serangan yang membuat sebuah *user* tidak dapat mengakses layanan internet yang dikarenakan melonjaknya *traffic* pada jaringan. Oleh karena itu, dibutuhkan suatu sistem yang dapat melakukan *feedback* terhadap serangan secara efisien.

Dalam tugas akhir ini, penulis akan membuat sebuah simulasi untuk melakukan sistem pencegahan terhadap serangan dengan menerapkan metode *traffic shaping* pada jaringan sebagai mekanisme pencegahan berbasis pada *router* yang dapat melakukan optimalisasi lau lintas jaringan dengan keluaran *rate limit bandwidth*. Menggunakan sistem berbasis *non agent* pada *router*. Analisis performansi jaringan juga dilakukan pada simulasi ini untuk mengetahuin nilai parameter quality of services (QoS).

Hasil pengujian dari metode traffic shaping terhadap quality of services yaitu melakukan standarisasi berdasarkan ITU-T. Sistem traffic shaping menggunakan metode *token bucket filter* mampu melakukan mitigasi terhadap serangan DDOS.

Kata kunci : DDOS, *non agent, traffic shaping, QoS, token bucket filter*

Abstract

Along with the development of technology at this point, needs of Internet access service as a medium of communication is increasing. This increase led to anomalies in network traffic. These anomalies, can occur because of a Distributed Denial of Service (DDoS) that deliberately or *Flashcrowd*, a large spike in network traffic because of the number of Internet *user*s who access the *server* rose significantly at a time. The impact of an anomaly is to make the *user* can not access the internet service.

In this final project, the author will make a simulation to perform preventive system against attacks by applying traffic shaping on the network as a prevention mechanism based on a router that can perform lau optimizing network traffic with the output rate limit bandwidth. Using a non-agent-based systems on the router. Network performance analysis is also performed on this simulation to know the value of the parameter of quality of services (QoS).

The test results from the method of traffic shaping on the quality of services that is standardized by ITU-T. System traffic shaping using the token bucket filter able to mitigate against DDOS attacks.

Keywords: DDOS, non agent, traffic shaping, QoS, Token bucket filter

1. Pendahuluan

Perkembangan teknologi internet yang begitu cepat menjadikan penyebaran informasi dan komunikasi menjadi tidak terbatas. Sehingga sisi keamanan suatu jaringan merupakan aspek penting yang harus diperhatikan. Tidak dapat dipungkiri bahwa saat ini pengguna internet sudah merata di berbagai kalangan masyarakat. Akses yang sangat mudah dalam mendapatkan informasi yang tersebar dengan luas dan cepat merupakan salah satu celah yang dapat dimanfaatkan oleh sebagian kalangan yang memanfaatkan kecanggihan teknologi internet. Banyak dari pengguna internet yang dirugikan karena adanya serangan-serangan yang tidak bertanggung jawab. Salah satu serangan yang dapat terjadi yaitu *Disributed Denial of Service* (DDoS) yang merupakan bentuk serangan *flooding* yang berusaha membuat suatu *host* atau *service* menjadi tak dapat diakses oleh user yang berhak. Sasaran serangan oleh DoS/DDoS adalah *link/bandwidth* untuk membuat sumber daya bandwidth penuh dan sumber daya komputasi pada server agar sistem pengolah kehabisan sumber daya yang berujung oleh jaringan *down* atau *crash* [2].

Berdasarkan penelitian [4], serangan DDoS sangat sulit dihentikan secara keseluruhan. Fokus dengan meminimalisir dampak dari serangan dan memaksimalkan perfomansi pada jaringan. Performansi dalam sebuah jaringan bergantung pada beberapa faktor seperti halnya konfigurasi jaringan, jumlah permintaan, dan metode manajemen jaringan tersebut.

Mengatur jalannya trafik merupakan salah satu cara yang dapat digunakan untuk mengatasi serangan tersebut. Pada *management traffic* dikenal dengan dua metode yaitu, *traffic shaping* dan *traffic policing*. Kedua metode tersebut dapat digunakan untuk mengatasi serangan.

Pada penelitian sebelumnya, telah dilakukan penerapan *traffic shaping* untuk melakukan *shaping bandwidth* pada jaringan. Penelitian tersebut hanya menghasilkan nilai *throughput*, namun belum dilakukan penelitian nilai q*uality of service* secara keseluruhan. Selain itu, penerapan metode traffic shaping hanya digunakan pada satu router saja. Oleh karena itu, dalam pada tugas akhir ini dilakukan analisis performansi jaringan terhadap penerapan metode *traffic shaping*, berdasarkan parameter *quality of service*.

2. Dasar Teori

2.1 Sistem Deteksi

Sistem deteksi serangan dikenal dengan dua istilah yaitu *Intrusion Detection System* (IDS) dan *Intrusion Prevention System* (IPS) [2]. Sistem Deteksi Intrusi atau IDS merupakan sebuah sistem yang mempunyai kemampuan untuk monitoring *traffic* jaringan, mendeteksi aktivitas-aktivitas yang mencurigakan, serta mampu melakukan pencegahan dini terhadap intrusi ataupun aktifitas yang dapat membahayakan sistem jaringan komputer [2]. IDS dapat dilakukan secara *collaborative* maupun *non-collaborative* [1]. *Collaborative* IDS bertujuan untuk mendeteksi serangan lebih akurat [2]. Gambar 2.1 menjelaskan bahwa *collaborative* IDS memiliki dua komponen, yaitu deteksi unit, dan korelasi unit yang dapat diklasifikasikan menjadi *centralized* IDS arsitektur [9]. Terdapat tiga komponen deteksi unit yang berada pada *low-level alert*, memberikan informasi atau laporan ke korelasi unit untuk dianalisis dan diambil tindakan selanjutnya [2].

2.2 Network Simulator

Network Simulator merupakan *software* simulasi yang berguna dalam mempelajari sifat dinamis jaringan komunikasi. NS2 terdiri dari dua bahasa utama : C++ dan *Object-oriented Tool Comand* (*OTcl*). Sementara C++ mendefinisikan mekanisme internal (yaitu, *backend*) simulasi. Bahasa Otcl membentuk simulasi dengan merakit dan mengkonfigurasi objek serta melakukan menjadwalan diskrit (yaitu, *Frontend*). C++ dan Otcl dihubungkan bersama menggunakan TclCL.

2.3 Non Agent

Sebuah sistem dikatakan sebagai agent-based ketika [1] agent memiliki suatu kemampuan sistem kecerdasan yang mampu berinteraksi (bekerja sama, berkoordinasi, negoisasi) dengan agent lain yang tersebar dalam suatu jaringan. Agent dapat mengambil keputusan dengan sendirinya dalam melakukan suatu tindakan, agent juga dapat memutuskan suatu tindakan atas permintaan agent lain [4]. Namun, penggunaan sistem berbasis agent dapat dilakukan ketika terdapat kasus dimana proses secara manual tidak bisa lagi dilakukan karena jumlah data yang besar, skala yang luas dan mahal [1]. Sistem berbasis non agent merupakan sistem yang tidak memiliki kecerdasan untuk mengambil keputusan dalam mengatasi suatu serangan dalam jaringan. Dalam ekseskusi suatu keputusan dapat dilakukan masih dubutuhkannya seorang administrator/owner jaringan itu sendiri [1]. Pemilihan sistem non agent dikarenakan jumlah node yang digunakan sedikit dengan skalabilitas yang tidak luas.

2.4 Bandwidth Management

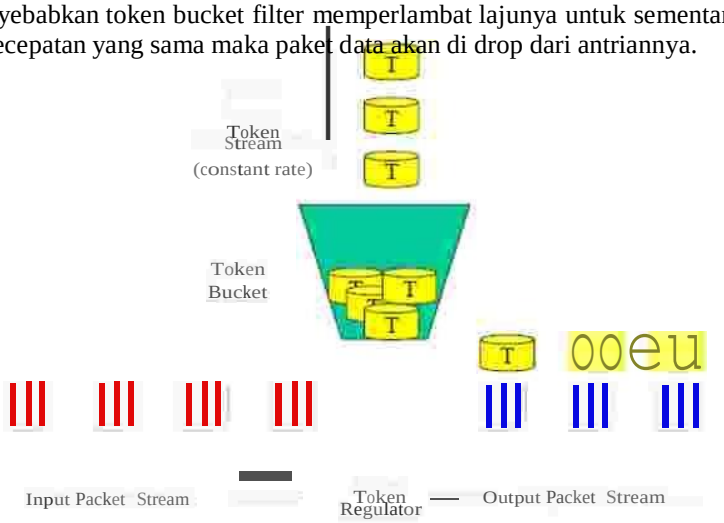
Bandwidth merupakan lebar pita frekuensi pada perangkat yang bias digunakan untuk transmisi data dalam komunikasi data. Di dunia komunikasi data terutama dalam jaringan TCP/IP, Bandwidth adalah sumber daya utama yang harus digunakan secara efesien. Dalam rangka pemanfaatan bandwidth secara optimal dan efisien maka diperlukan teknik khusus untuk memanajemen sumber daya tersebut. Sehingga dengan manajemen bandwidth, sumber daya jaringan bisa dikendalikan dan dapat memenuhi kriteria Quality of Service (QoS) [3]. *Bandwidth management* memiliki dua metode manajemen trafik yaitu *traffic shaping* dan *traffic policing* [3]. Secara sederhana, *traffic shaping* dapat diartikan dengan suatu metode manajemen trafik yang mengontrol *volume* trafik dengan periode yang spesifik dalam melakukan *rate limit*. Sedangkan *traffic policing* merupakan metode manajemen trafik yang melakukan *dropping* paket sesuai dengan aturan yang berlaku pada jaringan.

2.5 Token Bucket Filter

TBF merupakan disiplin antrian yang hanya melewatkan paket-paket datang dengan kecepatan tidak melebihi dari kecepatan yang telah ditentukan. TBF terdiri dari sebuah *buffer* (*bucket*), yang secara konstan terisi oleh beberapa informasi *virtual* yang dinamakan *token*, dengan kecepatan yang spesifik (*token rate*). Token bucket filter (TBF) membatasi bandwidth dengan metode shape & drop, prinsip kerja dari token bucket filter yaitu menggunakan aliran token

yang memasuki bucket dengan kecepatan (rate) yang konstan. Jika token dalam bucket habis maka paket data akan di antri dan jika antrian paket data melebihi dari kapasitas bucket maka paket data akan dibuang (drop). Setiap token yang datang menampung satu paket data yang ada pada antrian kemudian token dihapus dari bucket. Token dalam bucket akan lebih cepat habis jika aliran paket data melampaui kecepatan token memasuki bucket. Parameter paling penting dari bucket adalah seberapa besar ukurannya untuk menampung token. Token bucket filter bekerja berdasarkan 2 aliran, yaitu token dan data. Dari cara kerja sebelumnya maka didapatkan 3 kemungkinan skenario yang mungkin terjadi pada token bucket filter yaitu:

1. Paket data datang dengan kecepatan yang sama dengan kecepatan token. Sehingga paket data yang datang memiliki token yang sesuai dan melewati antrian tanpa delay.
2. Paket data datang dengan kecepatan yang lebih kecil daripada kecepatan token. Hanya sebagian dari token yang dihapus pada keluaran saat menampung paket data keluar dari antrian, sehingga sisa token yang ada terakumulasi memenuhi bucket. Sisa token yang tidak terpakai ini kemudian digunakan untuk mengeluarkan paket data pada kecepatan yang melebihi kecepatan standar token pada waktu yang singkat.
3. Paket data datang dengan kecepatan yang lebih besar daripada kecepatan token. Hal ini berarti bucket akan segera kehabisan token yang menyebabkan token bucket filter memperlambat lajunya untuk sementara waktu. Jika paket data terus datang dengan kecepatan yang sama maka paket data akan di drop dari antriannya.



Gambar 2.3 *Token Bucket Filter*

3. Pembahasan

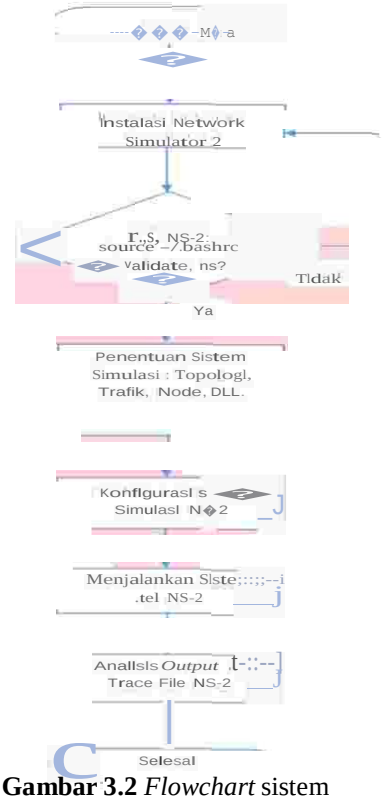
3.1 Deskripsi Sistem

Pada penelitian ini akan dilakukan manajemen trafik untuk mengatasi suatu serangan DDOS dengan menggunakan salah satu metode *traffic shaping* yaitu token bucket filter. Proses simulasi dan analisis dilakukan pada *software* Network Simulator 2. Hasil keluaran dari simulasi berupa data uji performansi berupa *trace file* yang digunakan untuk mengukur beberapa parameter quality of services yaitu delay, jitter, packet loss dan throughput.

Gambar 3.1 Blok Diagram Sistem

3.2 *Flowchart* Sistem

Pada penelitian tugas akhir ini mengikuti alur sistem seperti berikut :



Gambar 3.2 *Flowchart* sistem

3.3 Parameter Simulasi

Input dan *Output* merupakan elemen penting dalam melakukan simulasi. Parameter input digunakan untuk menghasilkan *output* yang berupa *trace file*. *Trace file* ini nantinya digunakan dalam mengukur *quality of service* yang berupa *throughput*, *delay*, *jitter*, maupun *packet loss*. Tabel 3.2 menjelaskan beberapa parameter *input* yang digunakan dalam simulasi.

Tabel 3.2 *Script Input* Simulasi

Script Code	**Deskripsi**
set ns [new Simulator]	*script* baru untuk memulai simulasi baru
set nf [open out.nam w] *$ns namtrace-all $nf*	Set NAM file terhubung dengan objek pada ns
set tr [open out.tr w] *$ns trace-all $tr*	Set Trace file pada simulasi
set n0 [$ns node]	Set node yang digunakan pada simulasi
$ns duplex-link $n0 $n1 10Mb 10ms DropTail	Set koneksi antar node dengan bandwidth
$ns duplex-link-op $n0 $n1 orient right-up	Set posisi node yang terhubung satu dengan lainnya.
set tcp [new Agent/TCP] *$ns attach-agent $n0 $tcp*	Set agent TCP untuk melakukan simulasi berbasis tcp
set udp [new Agent/UDP] *$ns attach-agent $n2 $udp*	Set agent UDP untuk melakukan simulasi berbasis tcp
$ns run	Menjalankan simualasi

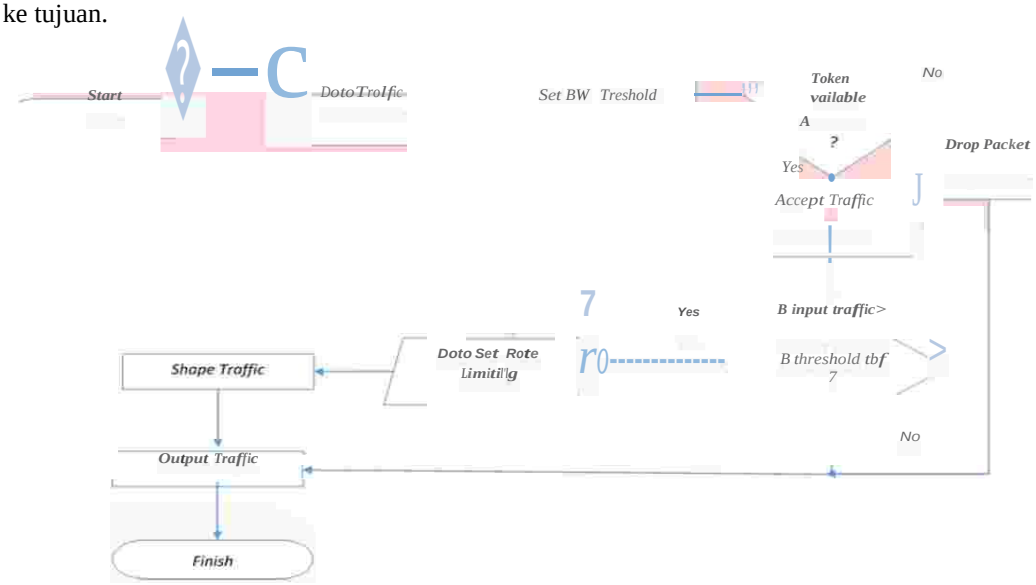
Output simulasi berupa trace file yang mencatat seluruh kejadian yang terjadi selama simulasi berlangsung. Format trace file dijelaskan melalui tabel berikut :

Table 3.3 *Format Trace file*

Event	*Time*	*From Node*	*To Node*	*Pkt type*	*Pkt size*	*Flags*	*Fid*	*Src ad*	*Dst ad*	*Seq num*	*Pkt id*

3.4 Traffic Shaping

Traffic shaping merupakan metode untuk melakukan *control* terhadap trafik dalam jaringan, khususnya dalam hal *bandwidth*. Menggunakan salah satu jenis metode *traffic shaping* yaitu *token bucket filter*. Pada gambar 3.5 dijelaskan bahwa tahap awal dalam melakukan uji sistem menggunakan metode *token bucket filter* yaitu *data traffic* untuk simulasi. Sistem bekerja dengan menggunakan *token*, jika *token* dalam *bucket* tersedia maka paket data akan di *queueing* dan jika antrian paket data melebihi dari kapasitas *bucket* maka paket data akan dibuang (*drop*). Selanjutnya trafik akan diproses sistem sesuai dengan *policy* yang berlaku. Jika *bandwidth* yang masuk melebihi *threshold*, maka *bandwidth* akan di *shape* sebelum sampai ke tujuan. Sedangkan untuk *bandwidth* yang masuk tidak melebihi *threshold*, maka trafik akan langsung diteruskan ke tujuan.

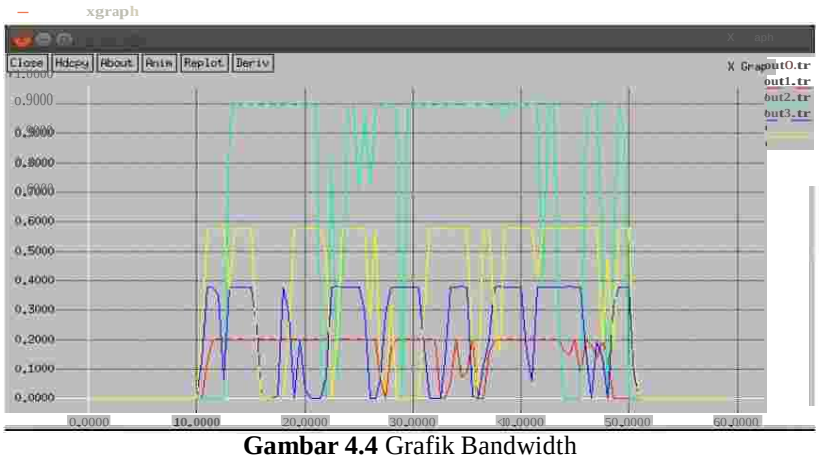


Gambar 3.5 *Flowchart Traffic Shaping*

4. Pengujian

4.1 Pengujian *Quality of Services*

Hasil percobaan dari empat skenario pengujian yang telah dilakukan mendapatkan *output* berupa *trace file* yang akan dianalisis. Terdapat *capture bandwidth* menggunakan *xgraph*, kemudian perhitungan *quality of service* yang terdiri dari *delay, jitter, packet loss* dan *throughput*. Berikut ini hasil dari pengujian dari skenario yang telah dilakukan:

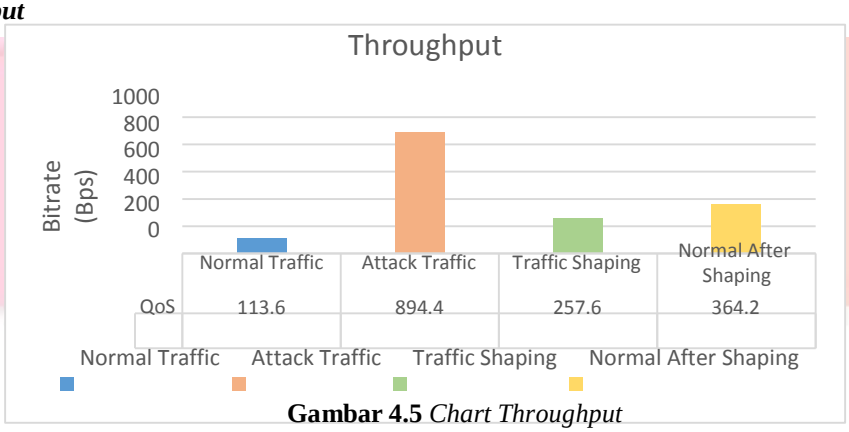


Gambar 4.4 Grafik Bandwidth

Gambar 4.4 menampilkan grafik warna merah merepresentasikan skenario pertama dengan keadaan *bandwidth* trafik dengan kondisi normal. Grafik warna hijau merepresentasikan skenario kedua dengan keadaan *bandwidth* trafik serangan tanpa dilakukan penanganan. Grafik warna biru merepresentasikan skenario ketiga dengan keadaan *bandwidth* trafik serangan yang telah dilakukan penanganan dengan *traffic shaping*. Grafik warna kuning merepresentasikan keadaan *bandwidth* normal setelah implementasi *traffic shaping* pada jaringan. Dari data grafik diatas, didapat hasil maksimum

bandwidth pada saat normal trafik yaitu 200,000 byte/s, trafik serangan 1,000,000 byte/s, setelah dilakukan penanganan serangan dengan *traffic shaping* menjadi 378,000 bytes/s dan *normal traffic after shaping* sebesar 578,000 bytes/s. Pengujian untuk analisis *quality of service* terhadap skenario pengujian telah dilakukan. Berikut ini merupakan hasil pengujiannya :

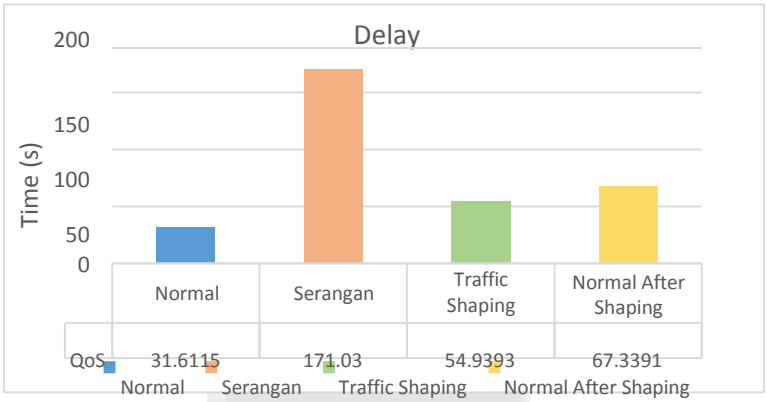
1. Throughput



Gambar 4.5 *Chart Throughput*

Pengujian *throughput* pada Gambar 4.5 menampilkan hasil percobaan yang dilakukan. Hasil pengujian didapatkan nilai *throughput* pada saat uji simulasi dengan trafik normal dengan nilai *throughput* sebesar 1758 paket yang sampai dengan kecepatan 113,6 bps dengan presentase paket sampai tujuan mencapai 100%. Pada *traffic* serangan nilai *throughput* mengalami peningkatan menjadi 12927 paket yang sampai ke tujuan dengan kecepatan *transfer* sebesar 894,4 bps. Hal ini dikarenakan adanya peningkatan trafik yang disimulasikan dengan menggunakan trafik serangan seperti pada skenario kedua mengakibatkan nilai *throughput* menjadi meningkat. Hasil pada skenario ketiga, nilai *throughput* menjadi 1173 paket yang diterima ke tujuan dengan kecepatan *transfer* sebesar 257,6 bps. Hal ini terjadi karena pada skenario ketiga telah diterapakan metode *traffic shaping* sehingga mempengaruhi nilai *throughput* pada skenario ketiga ini. *Rate bandwidth* pada *node* serangan telah di *shape* sehingga menghasilkan nilai *throughput* yang lebih rendah dengan kesuksesan paket sampai pada tujuan sebesar. Percobaan dari skenario keempat, menghasilkan nilai *throughput* normal setelah dilakukannya metode *traffic shaping*. Nilai *throughput* yang didapat yaitu 2340 paket ke tujuan dengan kecepatan *transfer* 364.2 bps. Hasil keempat ini merupakan *throughput* normal yang didapat dari percobaan perhitungan dengan trafik normal dan trafik serangan setelah dilakukannya *traffic shaping*.

2. Delay

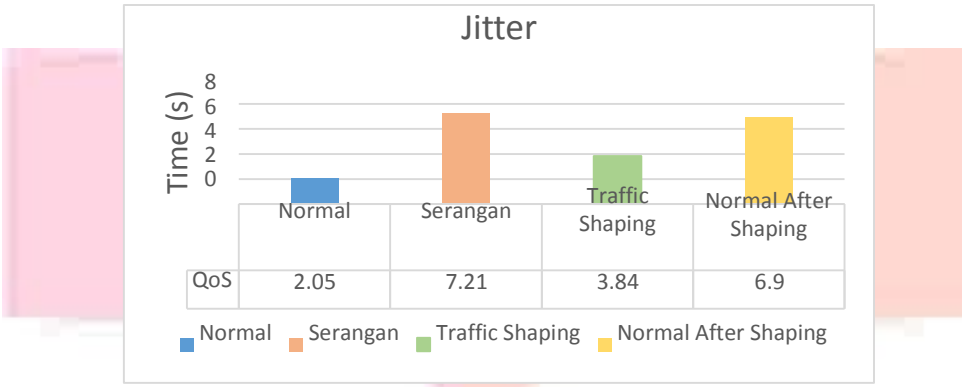


Gambar 4.6 *Chart Delay*

Hasil percobaan dari keempat skenario, telah dilakukan analisis dalam mencari *delay* yang terjadi pada jaringan. Skenario pertama yaitu percobaan dengan trafik normal didapat nilai *delay* 31,6115 s. Untuk simulasi dengan trafik serangan didapat nilai d*elay* sebesar 171,03 s. Dan untuk simulasi dengan trafik serangan yang telah diatasi dengan *traffic shaping* nilai *delay* yaitu 54,939 s. Nilai *delay* normal trafik setelah dilakukan metode *traffic shaping* sebesar 67,3391 s. Dari keempat hasil diatas, hasil pertama, ketiga dan keempat masih sesuai dengan standard ITU.T [15] dengan kategori *"acceptable"*. Namun untuk hasil skenario kedua, *delay* mencapai 171,03 s dimana nilai tersebut mendapatkan kategori

"*unacceptable*". Hal ini dikarenakan pada skenario kedua merupakan skenario serangan yang menyebabkan *delay* meningkat. Dari hasil ketiga dan keempat terhadap penggunaan *traffic shaping*, dapat dikatakan bahwa metode ini mampu dalam memitigasi serangan dengan menurunkan *delay* yang terjadi akibat serangan pada trafik.

3. Jitter



Gambar 4.7 *Chart Jitter*

Hasil percobaan yang telah dilakukan, nilai *jitter* pada saat uji simulasi dengan trafik normal didapat nilai *jitter* sebesar 2,05 s. Simulasi dengan trafik serangan didapat nilai *jitter* sebesar 7,21 ms. Simulasi dengan trafik serangan yang telah diatasi dengan *traffic shaping* nilai *jitter* yaitu 7,21 s. Hasil untuk trafik normal setelah *shaping* yaitu 6,9 s. Dari hasil percobaan diatas, terjadi peningkatan nilai *jitter* pada percobaan skenario kedua. Peningkatan tersebut terhadi karena trafik serangan. Namun, ketika digunakan metode *traffic shaping*, nilai *jitter* mengalami penurunan. Dengan hasil ini, *traffic shaping* berpengaruh terhadap nilai *jitter*, namun masih memiliki nilai *jitter* yang melebihi dari trafik normal. Hal ini diakibatkan karena adanya pengaruh proses penggunaan *token bucket filter* pada metode *traffic shaping router*.

4. Packet Loss

Pengujian menggunakan normal trafik, didapatkan hasil *packet loss* sebesar 8,23 %. Nilai ini dikarenakan adanya paket yang di *drop* efek dari serangan yang terjadi pada jaringan. Pengujian dengan trafik serangan, didapatkan *packet loss* sebesar 11,76 %. Trafik serangan melakukan pengiriman paket sebanyak 14650 paket, namun hanya sampai ketujuan sebesar 12927. Artinya sebanyak 1723 paket telah di *drop*. Hal ini yang menyebabkan packet loss pada trafik serangan menjadi 11,76%. Kemudian, pada pengujian penggunaan *traffic shaping* dan *normal after shaping* didapatkan nilai *packet loss* sebesar 0%. Nilai ini didapatkan karena keberhasilan seluruh paket yang dikirim mencapai target tanpa adanya paket yang di *drop*.

Tabel 4.3 Hasil *Packet Loss*

Normal	Serangan	*Traffic shaping*	*Normal After Shaping*
8,23 %	11,76 %	0 %	0 %

5. Kesimpulan dan Saran

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari Tugas Akhir ini adalah :

1. Pemilihan metode *traffic shaping* yang tepat dapat digunakan sebagai langkah pencegahan atau mitigasi terhadap serangan.
2. Pada pengujian performansi jaringan berdasarkan *quality of service*, pada skenario pertamad dengan trafik normal didapatkan nilai throughput 2246 bps, delay 14.62 ms, jitter 4.05 ms , packet loss 0 %.
3. Pada pengujian performansi jaringan berdasarkan *quality of service*, pada skenario kedua dengan trafik serangan didapatkan nilai throughput 29890 bps, delay 49.81 ms, jitter 13.46 ms , packet loss 0 %.
4. Pada pengujian performansi jaringan berdasarkan *quality of service*, pada skenario ketiga dengan trafik serangan setelah diatasi dengan *traffic shaping* didapatkan nilai throughput 4857 bps, delay 31.625 ms, jitter 8.75 ms , packet loss 0 %.
5. Penggunaan metode *traffic shaping* terbukti dapat melakukan mitigasi terhadap serangan yang terjadi pada jaringan. Hal ini dapat dibuktikan dari beberapa hasil percobaan yang telah dilakukan.

6. Penggunaan metode *traffic shaping* dalam mengatasi serangan ddos dapat menyebabkan delay dan jitter lebih tinggi dibandingkan pada saat trafik normal, namun dapat dengan efektif menurunkan lonjakan bandwidth akibat adanya serangan yang masuk. Efek terjadinya delay dan jitter yang lebih besar ketika terjadi serangan pun dapat diatasi oleh metode ini.

5.2 Saran

Saran untuk penelitian selanjutnya adalah :

1. Terdapat adanya kondisi delay dan jitter yang tinggi pada saat penggunaan metode ini, namun hal itu masih dalam batas wajar sesuai dengan standar ITU-T. Selain itu, metode ini dapat menurunkan lonjakan bandwidth akibat adanya serangan. Diharapkan untuk kedepannya dikembangkan dengan menggunaka metode lain dalam melakukan *management traffic* untuk mengatasi serangan.
2. Untuk penelitian ini masih sebatas simulasi. Diharapkan untuk kedepannya dapat digunakan dalam kondisi *real* pada jaringan komputer.
3. Penggunaan metode lain dalam mengatasi serangan sangat dianjurkan untuk perkembangan penelitian kedepaannya.

DAFTAR PUSTAKA

[1] K. MALIALIS, "Distributed Reinforcement Learning for Network Intrusion Response," 2014.

[2] Yudha Purwanto, Kuspriyanto, Hendrawan, dan Budi Rahardjo, "Traffic Anomaly Detection in DDoS Flooding Attack," in *THE 8TH INTERNATIONAL CONFERENCE ON TELECOMMUNICATION SYSTEM,SERVICES,AND APPLICATION*, 2014.

[3] Cisco, Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2 Cisco, 2014.

[4] Christos Douligeris & Aikaterini Mitrokotsa, "DDoS attacks and defense mechanisms: classification," 2003.

[5] Teerawat Issariyakul and Ekram Hossain , Introduction to network simulator NS2, Springer Science & Business Media, Edition 2009.

[6] Seitz, N., NTIA/ITS, " ITU-T QoS Standards for IP-Based Networks," *IEEE Communications Magazine.*, 2003.

[7] Kumar, Vinod. "Congestion Control Techniques Quality Of Services". Study Material BCA, MCA & FUN. N.p., 2012. Web. 24 June 2016.

[8] Li-Chiou Chen, Kathleen Carley, "Modeling Distributed Denial Of Services Attack and Defense," 2002.

[9] Yu Chen, Kai Hwang, Wei-Shinn Ku, "Collaborative Detection of DDoS Attacks over Multiple Network Domains".

[10] Ming Li, Jun Li, Wei Zhao, "Experimental Study of DDOS Attacking Flood Type Based on NS 2," 2009.

[11] H. R. Rabiee, "*Traffic* Access Control," Digital Media Lab - Sharif University of Technology, 2012

[12] Markus Goldstein, Matthias Reif, Thomas Breuel, "High Performance *Traffic shaping* for DDoS Mitigation," 2008.

[13] Shiv Kumar, Ritika Singal, Priyadarshn, "Mitigate the Impact of DoS Attacks by Verifying Packet Structure," in *International Journal of Advanced Research in Computer Science and Software Engineering*, Kalan, India, 2013.

[14] Vooka Pavan Kumar, Abhinava Sundaram P, Munnaluri Bharath Kumar, N.Ch.S.N.Iyengar, "ANALYSIS OF DDoS ATTACKS IN DISTRIBUTED PEER TO PEER NETWORKS," vol. 2, 2011.

[15] Islam Hegazy, Hossam Faheem, "A Multi-agent Based System for Intrusion Detection," 2003