INTEGRASI *INTRUSION PREVENTION SYSTEM* DAN ANALISA PERFORMANSI PADA *SOFTWARE DEFINED NETWORK*

INTRUSION PREVENTION SYSTEM INTEGRATION AND PERFORMANCE ANALYSIS ON SOFTWARE DEFINED NETWORK

[1] Pande Putu Kika Adi Saputra [2] Ridha Muldina Negara, S.T., M.T. [3] Danu Dwi Sanjoyo, S.T., M.T.
[1,2,3]Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom
Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia
[1] kika.adis@gmail.com [2] ridhanegara@telkomuniversity.ac.id [3] danudwj@telkomuniversity.ac.id

Abstrak

***Software Defined Network* merupakan sebuah arsitektur jaringan yang memisahkan fungsi *controlling* dan fungsi *forwarding* sebuah perangkat jaringan, intinya adalah sentralisasi jaringan. Namun konsep jaringan yang tersentralisasi memiliki kelemahan dari segi keamanan *control plane*. Pada Tugas Akhir ini bertujuan menerapkan dan menganalisa pengaruh mekanisme keamanan dari segi performansi menggunakan *Intrusion Prevention System* (IPS) yang memiliki keunggulan dengan adanya mekanisme pemblokiran paket sebagai fungsi preventif.**

Pada penelitian ini telah dilakukan integrasi IPS pada jaringan SDN serta melakukan pengujian dan pengukuran terhadap pengaruh yang ditimbulkan dari segi performansi jaringan. Parameter yang digunakan adalah QoS antara lain *delay*, *jitter*, *throughput*, serta *packet loss ratio*. Selain itu juga dilakukan pengujian untuk mengukur batas ketahanan IDS Snort terhadap serangan dalam jumlah banyak.

Dari pengujian yang telah dilakukan, jaringan SDN yang terintegrasi IPS dilihat dari hasil pengujian performansi parameter QoS cenderung lebih stabil, karena mampu memblokir paket serangan sehingga meminimalisir terjadinya penurunan performansi akibat adanya paket serangan yang masuk ketika pengiriman paket layanan. Selain itu kemampuan IPS dalam menganalisa paket yang masuk mengalami penurunan ketika jumlah paket serangan memasuki 9.000 paket/s.

Kata Kunci: *Software Defined Network* (SDN), *Intrusion Prevention System* (IPS), *Snort*, Performansi, Ryu

Abstract

Software Defined Network is a network architecture that separates the functions of controlling and forwarding function of a network device, essentially is network centralization. However, the concept of a centralized network has a weakness in terms of security control plane. In this Final Project aims to apply and analyze the influence of security mechanisms in terms of performance using Intrusion Prevention System (IPS) which has advantages in the presence of packet blocking mechanism as a preventive function.

In this study integration of IPS on the SDN network has been conducted and testing and measuring the effects caused in terms of network performance. The parameters used are QoS including delay, jitter, throughput, and packet loss ratio. In addition, testing is also done to measure the limits of IDS Snort resistance to attacks in large quantities.

From the tests that have been done, SDN Network integrated with IPS is able to block the attack packets sent simultaneously with the service pack of video stream and VoIP. SDN network integrated with IPS seen from the results of QoS parameter performance testing tend to be more stable, because it is able to block the attack packets to minimize the decrease in performance due to the incoming packet attacks when the delivery of service packets.. In addition, the ability of IPS in analyzing incoming packets decreased when the number of attack packets entered 9,000 packets / s.

Keywords: Software Defined Network (SDN), Intrusion Prevention System (IPS), Snort, Performance, Ryu

1. Pendahuluan

Seiring berkembangnya kemampuan sebuah jaringan, mengkonfigurasi perangkat jaringan satu demi satu dianggap tidak efisien, terutama pada jaringan yang memiliki perangkat dalam jumlah banyak. Konsep *Software Defined Network* (SDN) yang memisahkan *control plane* dan *data plane* yang mana hanya satu perangkat yang berfungsi sebagai *control plane*, sedangkan perangkat jaringan lain hanya memiliki *data plane*, sehingga perangkat tersebut hanya memiliki fungsi *forwarding*. Hal ini dipandang lebih efektif dan efisien bagi administrator jaringan dalam memanajemen jaringan tersebut. Namun konsep jaringan yang tersentralisasi tentunya memiliki kelemahan dari segi keamanan *control plane* itu sendiri.

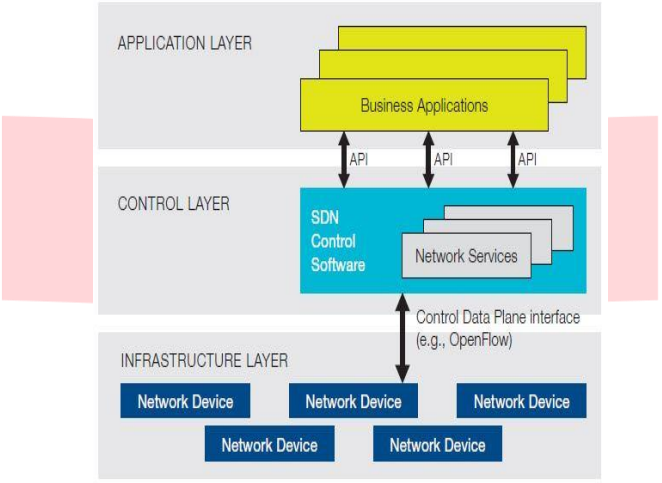
Mekanisme keamanan pada SDN pada dasarnya masih dikembangkan, salah satu penelitian yang mengembangkan mekanisme keamanan pada SDN berjudul "Integrasi *Intrusion Detection System* pada *Sofware Defined Network*", berdasarkan ide dari penelitian tersebut, pada Tugas Akhir ini menerapkan mekanisme keamanan dengan menggunakan *Intrusion Prevention System* (IPS) pada SDN. IPS pada dasarnya merupakan IDS yang dikombinasikan dengan *Firewall*, sehingga dapat melakukan fungsi preventif.

Pada penelitian kali ini akan dibangun sebuah jaringan sederhana dengan konsep SDN yang akan diintegrasikan dengan IPS, menggunakan kontroler *Ryu*. Integrasi IPS pada SDN diharapkan mampu meningkatkan peforma jaringan dari segi keamanan.

2. Landasan Teori

2.1 Software Defined Network (SDN)

Software Defined Network (SDN) merupakan sebuah konsep jaringan yang menggunakan arsitektur berbeda dengan arsitektur jaringan konvensional. Pada SDN *control plane* dipisahkan dengan *forwarding plane* [2], yang mana pada arsitektur jaringan konvensional fungsi *forwarding plane* dan fungsi *control plane* terdapat dalam satu perangkat. Agar perangkat jaringan yang sudah tidak memiliki fungsi *controlling* dapat menerima instruksi dari *controller*, dibutuhkan sebuah protokol yang menjadi perantara antara *control layer* dan *infrastructure layer* seperti ditunjukkan oleh "Gambar 2.1".

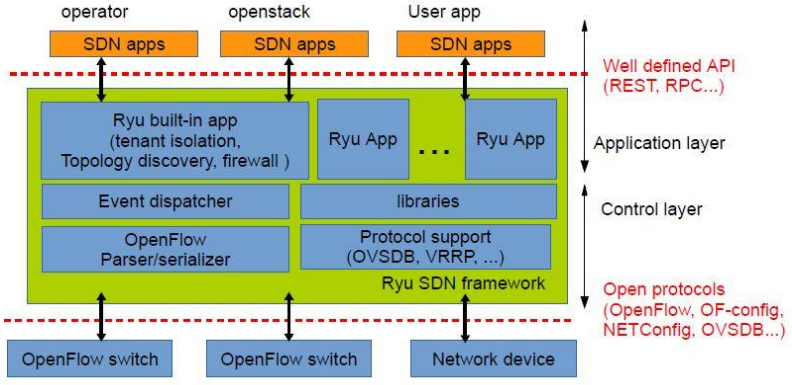


Gambar 2. 1 Arsitektur SDN [4]

2.2 Ryu *Controller*

Ryu adalah salah satu *controller* yang dapat digunakan pada jaringan SDN, *controller* ini menggunakan Python sebagai bahasa pemrogramannya. Selain mendukung penggunaan protokol OpenFlow, Ryu juga mendukung penggunaan protokol lain seperti: Netconf, OF-config, SNMP. Seperti ditunjukkan pada "Gambar 2.3".

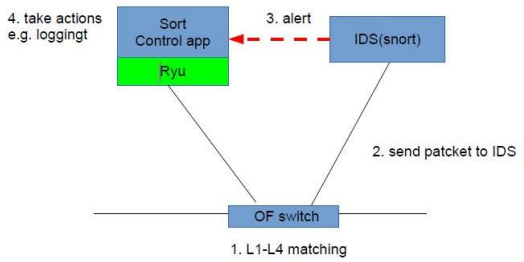
Ryu didefinisikan antara *application layer* dan *infrastructure layer*, Ryu juga memiliki aplikasi bawaan yang dapat dimanfaatkan, ditunjukkan pada arsitektur Ryu pada "Gambar 2.3".



Gambar 2. 2 Arsitektur SDN [11]

2.2.1 Integrasi Snort

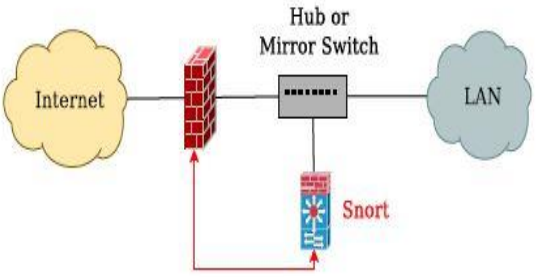
Salah satu fitur Ryu yang akan dimanfaatkan dalam Tugas Akhir kali ini adalah Integrasi Snort. Dengan Integrasi Snort, Ryu dapat menindaklanjuti *alert* yang dikirimkan oleh Snort, seperti ditunjukkan pada "Gambar 2.4".

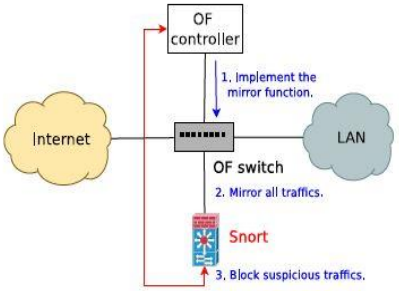


Gambar 2. 3 Alur kerja Integrasi Snort [11]

2.3 Snort

Snort adalah IPS *open source*, yang mampu melakukan analisis *traffic* secara *real-time* dan *packet logging* pada jaringan IP [9]. Pada arsitektur tradisional Snort bekerja dengan cara menduplikat semua paket yang lewat ke Snort, kemudian dicocokkan dengan rules yang terdapat pada Snort, jika ada paket yang mencurigakan maka Snort akan menyalakan *alert*, seperti ditunjukkan pada "Gambar 2.8".





Gambar 2. 4 Cara kerja Snort pada jaringan tradisional [14] Gambar 2. 5 Cara kerja Snort pada jaringan SDN [14]

Pada jaringan SDN, Snort menerapkan cara kerja yang berbeda, seperti ditunjukkan "Gambar 2.9", yaitu dengan menerapkan *mirror function* pada *switch* OpenFlow. *Controller* akan mengatur satu port *input* dengan dua port *output*, satu port untuk *forwarding*, dan port *output* lainnya sebagai port untuk Snort. Paket yang lewat akan diteruskan ke tujuan sekaligus ke *snort* untuk dianalisis. Jika Snort menemukan paket yang mencurigakan makan Snort akan mengirim *alert* ke *controller* untuk menjalankan mekanisme pemblokiran paket.

2.4 *Denial of Service* (DoS)

DoS merupakan salah satu jenis serangan pada jaringan komputer yang yang bertujuan menyerang *availability* dari target yang dituju, biasanya *server* ataupun *user* dalam sebuah jaringan. DoS dilakukan dengan cara membanjiri target dengan jumlah data yang sangat banyak, sehingga menghabiskan sumber daya (*resource*) yang diimiliki hingga tidak dapat menjalankan fungsi dengan benar, secara tidak langsung mengakibatkan *user* lain tidak dapat mengakses layanan dari target yang diserang.

3. Perancangan dan Implementasi

3.1 Desain Sistem

Secara umum, sistem yang dibuat adalah sebuah jaringan SDN, dilengkapi dengan sistem keamanan IDS Snort sekaligus mekanisme penanganan serangan yang terintegrasi dengan *controller* SDN. Integrasi Ryu dan Snort akan ditempatkan pada salah satu komponen jaringan. Sistem akan diuji dengan beberapa metode serangan *cyber*, ketika paket yang dikirim oleh 'penyerang' melewati sistem keamanan IPS, paket tersebut akan dicocokkan dengan *rules* yang sudah dipasang pada Snort. Jika paket dianggap berbahaya, maka sistem akan melakukan tindakan yang mencegah paket untuk masuk ke dalam jaringan.

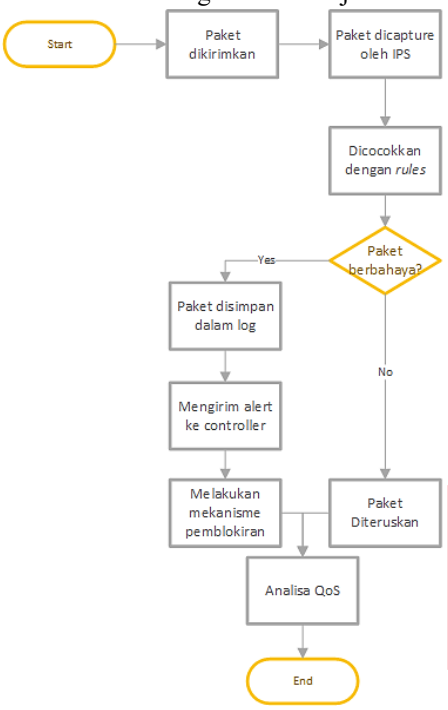
3.1.1 Alur Kerja Sistem

Pertama-tama paket akan dikirimkan tanpa ada paket serangan, setelah beberapa saat paket yang berasal dari 'penyerang' akan mulai dikirimkan.

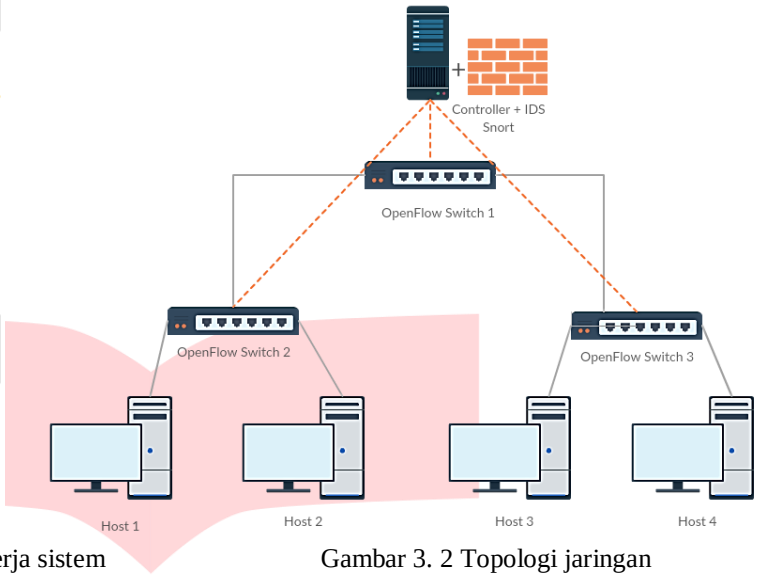
Ketika paket melalui mekanisme keamanan, maka paket akan di-*capture* oleh IPS, paket yang di-*caputre* akan dicocokkan dengan *rules* yang sudah dipasang pada IPS. Jika paket tidak cocok dengan *rules* yang dipasang, maka menandakan tidak adanya tanda paket berbahaya, paket akan diteruskan dan akan dilakukan analisa QoS.

Jika paket yang di-*capture* cocok dengan salah satu *rules* yang dipasang, maka paket akan disimpan dalam *log*, sistem akan mentriger *alert* yang akan dikirimkan ke *controller*. Kemudian *controller* akan menjalankan mekanisme pemblokiran untuk mencegah paket masuk ke jaringan. QoS akan tetap dianalisa

apakah terjadi perubahan pada parameter yang telah ditetapkan setelah dilakukan mekanisme pemblokiran. Untuk diagram alur kerja sistem terdapat pada "Gambar 3.1".



Gambar 3. 1 Diagram alur kerja sistem



Gambar 3. 2 Topologi jaringan

3.1.2 Rancangan Topologi Sistem

Topologi jaringan dari sistem yang akan digunakan pada sistem yang akan dibangun adalah sebagai berikut:

Pada "Gambar 3.2" menunjukkan topologi yang akan disimulasikan. Topologi ini merepresentasikan jaringan SDN sederhana untuk melakukan uji performansi. Komponen–komponen pada topologi tersebut adalah sebagai berikut:

1. *Controller*
   Komponen ini berfungsi mengatur empat *switch* yang ada pada topologi tersebut. *Controller* yang digunakan adalah Ryu yang dijalankan degan aplikasi *built-in* Firewall pada Ryu. *Controller* ini akan terintegrasi dengan Snort untuk melakukan mekanisme pertahanan.
2. IDS Snort
   Komponen ini berfungsi sebagai mekanisme deteksi terhadap paket yang masuk dan mengirimkan *alert* kepada *controller* jika paket dianggap berbahaya.
3. OpenFlow *Switch*
   Komponen ini merupakan *switch* yang sudah mendukung teknologi OpenFlow. Masing-masing *switch* terhubung ke *controller.*
4. *Host*
   Komponen ini adalah *host* yang masing-masing akan terhubung ke OF*Switch.*

3.2 Desain Perangkat Keras

Pada sistem kali ini akan digunakan satu buah laptop, menjalankan VM sebagai *controller,* IDS, dan emulator jaringan. Untuk alamat IP pada jaringan ini berada dalam satu subnet. Untuk pemetaan alamat IP dapat dilihat pada "Tabel 3.1".

Tabel 3. 1 Pemetaan alamat IP

Komponen	Alamat IP
Host 1	10.0.0.1
Host 2	10.0.0.2
Host 3	10.0.0.3
Host 4	10.0.0.4
Controller	127.0.0.0.1:6653

3.3 Desain Perangkat Lunak

Perangkat lunak yang akan digunakan pada sistem kali ini adalah sebagai berikut:

1. Linux Ubuntu 14.04 LTS pada emulator jaringan, *controller*, IDS, dan *server*.
2. Linux Kali sebagai penyerang.
3. Mininet sebagai emulator jaringan SDN.
4. Ryu sebagai *controller.*
5. Snort sebagai IDS.
6. *Distributed Internet Traffic Generator* (D-ITG) sebagai generator paket data, video, dan VoIP.
7. Iperf untuk melakukan pengaturan *background traffic.*
8. Bahasa pemrograman *Python* untuk konfigurasi topologi jaringan pada Mininet.
9. *Born Again Shell* (Bash) untuk skrip pemblokiran.
10. Hping untuk melakukan serangan DOS Syn *flood*, ICMP *flood.*

3.4 Perencanaan Pengujian Sistem

Pada penelitian kali ini pengujian berfokus pada dua hal, yaitu membandingkan tingkat performansi jaringan SDN sebelum dan sesudah terintegrasi dengan IPS dalam kondisi sama-sama menerima serangan ketika mengirimkan paket layanan, serta mengetauhi sejauh mana batasan IPS mampu menangani serangan yang masuk. Skenario yang akan digunakan pada penelitian kali ini meliputi skenario uji performansi serta skenario uji ketahanan IPS.

3.4.1 Skenario Uji Performansi

Uji performansi menggunakan QoS sebagai parameter pengujian, parameter yang digunakan antara lain adalah *delay*, *throughput*, *jitter*, dan *packet loss ratio ratio*. Pada uji performansi ini, *delay* yang digunakan adalah *one way delay*.

a. Tujuan Pengujian

Pengujian dilakukan untuk mengetahui pengaruh integrasi IPS pada jaringan SDN dari segi performansi jaringan. Parameter pengujian yang digunakan adalah QoS antara lain *delay*, *throughput*, *jitter*, dan *packet loss ratio ratio*. Pada uji performansi ini, *delay* yang digunakan adalah *one way delay*.

b. Sistematika Pengujian

Pengujian dilakukan dengan mengirimkan dua jenis *traffic* UDP [7] yang dibangkitkan menggunakan *packet generator* D-ITG. *Traffic* yang dibangkitkan antara lain:

1. Video *streaming* 24 frame per detik, yang mana satu *frame* sama dengan satu paket yang menggunakan distribusi normal μ =27791 bytes dan σ^2 =6254 bytes.
2. VoIP menggunakan codec G.711 sebanyak 100 paket per detik, ukuran paket 80 bytes, tanpa menggunakan *Voice Activation Detection* (VAD).

Dalam pengujian ini juga membangkitkan *background traffic* menggunakan Iperf dengan nilai *background traffic* sebesar 25 Mbps, 75 Mbps, kemudian 90 Mbps dinaikkan 1 Mbps secara bertahap hingga 100 Mbps, untuk mengetahui pada titik mana terjadi perubahan performansi jaringan secara signifikan. Selain itu untuk mengetahui karakteristik dari IPS, ketika mengirim paket layanan sistem diuji dengan dua skenario serangan:

1. Skenario 1

   DoS *Syn flood*, paket TCP yang hanya memuat flag *syn*, ukuran paket 44 bytes, dikirim dengan mode –flood (dikirim dengan interval secepat mungkin)
2. Skenario *2*

   DoS dengan paket ICMP, ukuran paket 1 Kbytes, dikirim dengan interval antar paket 1 µs.

.

3.4.2 Skenario Uji ketahanan IDS Snort

Uji ketahanan IDS menggunakan *detection rate* pada snort untuk melihat seberapa batasan IDS untuk menangani serangan dalam jumlah banyak.

a. Tujuan Pengujian

Pengujian dilakukan untuk melihat seberapa batasan IDS untuk menangani serangan dalam jumlah banyak.

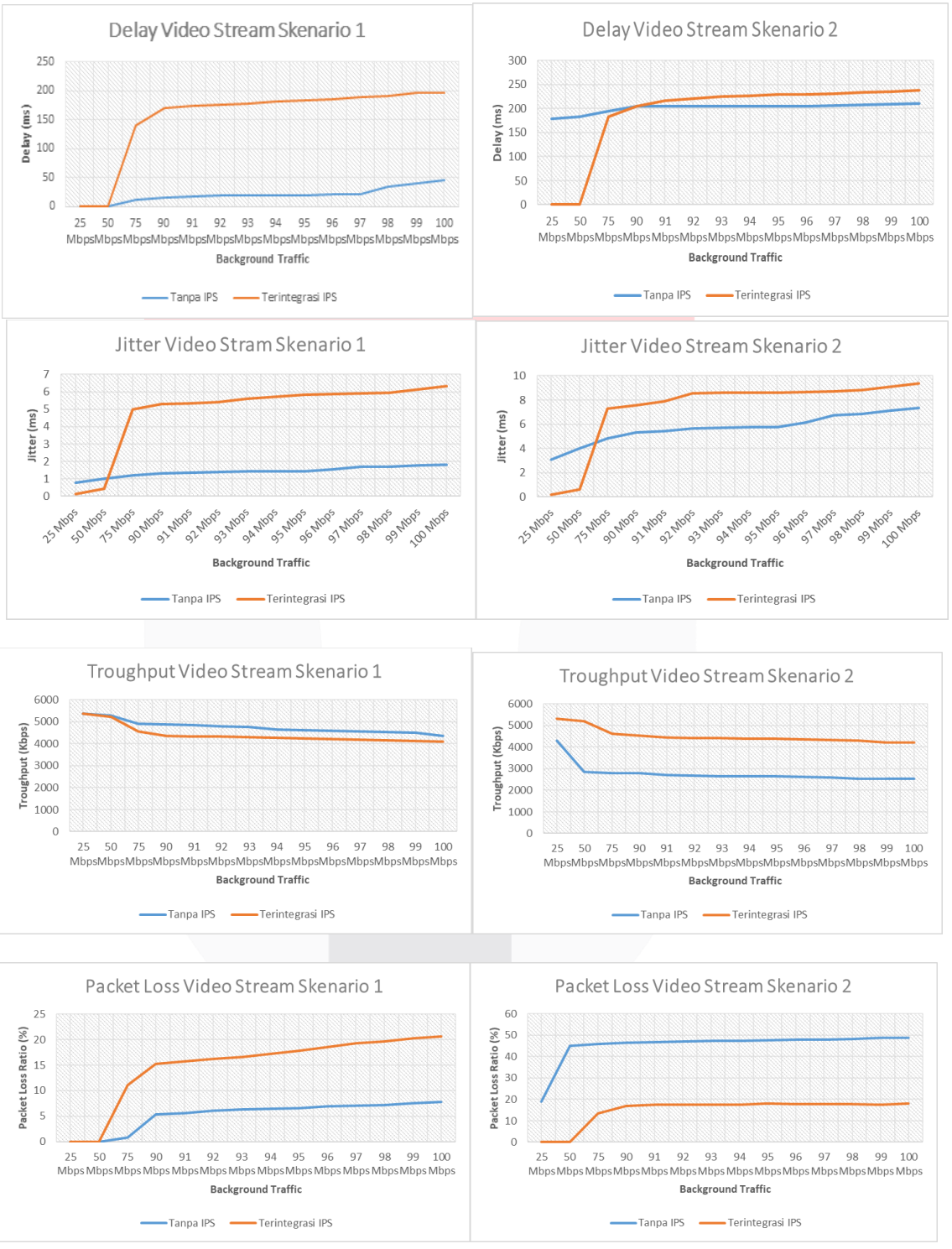
b. Sistematika Pengujian

Pengujian dilakukan menggunakan *detection rate* IDS Snort sebagai parameter pengujian. Untuk mengirimkan paket serangan menggunakan hping3, dengan mengirimkan paket dalam jumlah yang ditingkatkan secara bertahap yaitu sebesar 1000 paket/s hingga 10.000 paket/s, kemudian ditingkatkan menjadi 15.000 paket/s, 25.000 paket/s, 50.000 paket/s, hingga 75.000 paket/s. Peningkatan *attack rate* dilakukan untuk mengetahui pada titik mana IDS Snort mulai mengalami penurunan dalam mendeteksi paket serangan yang masuk.

4. Pengujian dan Analisis Implementasi Sistem

4.1 Analisis Skenario Uji Performansi

Uji performansi menggunakan QoS sebagai parameter pengujian, parameter yang digunakan antara lain adalah *delay*, *throughput*, *jitter*, dan *packet loss ratio ratio*.



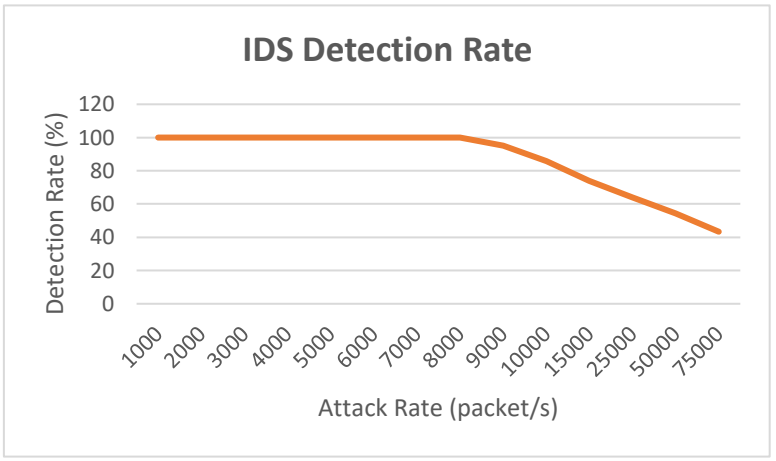
Gambar 4.1 Hasil uji performansi jenis trafik paket *Video Stream*

Gambar 4.2 Hasil uji performansi jenis trafik VoIP

Jaringan SDN yang terintegrasi IPS dilihat dari hasil pengujian performansi parameter QoS cenderung lebih stabil, karena mampu memblokir paket serangan sehingga meminimalisir terjadinya penurunan performansi akibat adanya paket serangan yang masuk ketika pengiriman paket layanan

4.2 Analisis Skenario Uji Ketahanan IDS

Pengujian dilakukan menggunakan *detection rate* IDS Snort sebagai parameter pengujian. Untuk mengirimkan paket serangan menggunakan hping3, dengan mengrimkan paket dalam jumlah yang ditingkatkan secara bertahap.



Gambar 4.3 Grafik IDS *Detection Rate*

Berdasarkan hasil pengujian pada "Gambar 4.8" dapat dilihat bahwa ketika *attack rate* 1.000 paket/s hingga 8.000 paket/s IDS masih mampu menganalisa paket dengan kondisi 100%, ketika attack rate memasuki 9.000 paket/s, IDS mulai mengalami penurunan dalam mengalisa paket yang masuk sehingga hanya mampu mendeteksi 95,13% paket yang masuk. Penurunan performa IDS dalam menganalisa paket semakin turun secara signifikan ketika jumlah paket yang dikirimkan ditingkat kelipatan 5.000 paket/s. Hal ini diakibatkan oleh keterbatasan *resource* yang mengakibatkan snort men-*drop* paket yang masuk ketika *detection engine* sebagai penganalisa paket sudah tidak mampu menganalisa paket dalam jumlah besar.

5. Kesimpulan

Berdasarkan analisa dari hasil pengujian skenario performansi dan uji ketahanan IDS pada sistem yang telah dibangun maka dapat diambil kesimpulan sebagai berikut:

1. Jaringan SDN yang terintegrasi IPS dilihat dari hasil pengujian performansi parameter QoS cenderung lebih stabil, karena mampu memblokir paket serangan sehingga meminimalisir terjadinya penurunan performansi akibat adanya paket serangan yang masuk ketika pengiriman paket layanan.

2. Kemampuan IPS Snort dalam menganalisa pake serangan yang masuk mulai mengalami penurunan pada 9.000 paket/s, diakibatkan oleh keterbatasan *resource* yang mengakibatkan snort men-*drop* paket yang masuk ketika *detection engine* sebagai penganalisa paket sudah tidak mampu menganalisa paket dalam jumlah besar.

Daftar Pustaka :

[1] Satria Akbar Mugitama, *Analisa Performansi Penanganan Kegagalan Link pada Layer 2 pada Jaringan Software Defined Network Openflow*. Bandung, 2015.
[2] Mininet Team, "Mininet Overview," [Online]. Available: http://mininet.org/overview/ . [Accessed 16 November 2017].
[3] Taufik Nur Fauzi, *Integrasi Intrusion Detection System pada Software Defined Network*. Bandung. 2016.
[4] Alamsyah, *Implementasi Keamanan Instrusion Detection System (Ids) Dan Instrusion Prevention System (Ips) Menggunakan Clearos*.
[5] Roesch, Martin. "*Snort: Lightwegiht intrusion detection for networks*." *Lisa*. Vol.99.No.1.1999.
[6] Nadeau, Thomas D., and Ken Gray. *SDN: Software Defined Networks: An Authoritative Review of Network Programmability Technologies*. "O'Reilly Media, Inc.", 2013.
[7] Nippon Telegraph and Telephone Corporation "Snort Integration" [Online]. Available: http://ryu.readthedocs.io/en/latest/snort_integrate.html. [Accessed 17 November 2017].
[8] Chi, Po-Wen, et al. "An AMI threat detection mechanism based on SDN networks." Proc. SECURWARE. 2014.
[9] Rafeeq Ur Rehman, *Intrusion Detection System with Snort, Advanced IDS Techniques Using Snort, Apache, MySQL, PHP, and ACID*. New Jersey, United States of America: Prentice Hall PTR, 2003.
[10] Telecommunication Standardization of ITU, "ITU-T Rec. G.1010," *Transmision Systems and Media, Digital Systems and Networks* , November 2011
[11] Stefano Avallone, Donato Emma, Antonio Pescape, and Giorgio Ventre, "A Practical Demonstration of Network Traffic Generation," in Eighth IASTED International Confrerence INTERNET AND MULTIMEDIA SYSTEMS AND APPLICATIONS, Kauai, Hawaii, USA, 2004, pp. 138-143.