Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)

journal homepage: http://journal.lembagakita.org/index.php/jtik

Simulasi Keamanan Jaringan pada DPDK *OpenvSwitch* Berbasis *Network-Based Intrusion Detection System* (NIDS)

Juan Tamalaki Ohyver [1*], Dian W. Chandra [2]

[1*] *Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.*

[2] *Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.*

a r t i c l e i n f o

Article history:
Received 24 November 2022
Received in revised form
25 February 2023
Accepted 1 May 2023
Available *online* July 2023

DOI:
https://doi.org/10.35870/jtik.v7i3.845

Keywords:
DPDK; OpenvSwitch; NIDS; Snort; Denial of Service; Network Security.

Kata Kunci:
DPDK; OpenvSwitch; NIDS; Snort; Denial of Service; Keamanan Jaringan.

a b s t r a c t

Virtual networks in a virtualized server environment use Virtual Switches to interconnect network traffic on the Virtual Machine with the physical network. However, at this time attacks carried out by hackers often occur in physical and virtual server environments. Hacking on computer network security is often done by a group of people who want to penetrate the security of a system. This activity aims to find, obtain, change, and even delete information. Therefore the use of OpenvSwitch and the application of NIDS in it is very useful for virtual network security. In this study, a simulation was carried out to find out how big the percentage of NIDS success was in detecting DOS attacks directed at OpenvSwitch. From the results of this study, the percentage of success for Snort in detecting attacks in this test is around ± 75%.

a b s t r a k

Jaringan virtual dalam lingkungan Server yang tervirtualisasi menggunakan Virtual Switch untuk menghubungkan lalu lintas jaringan pada Virtual Machine dengan jaringan fisik. Akan tetapi pada masa ini serangan yang dilancarkan oleh peretas seringkali terjadi pada lingkungan Server fisik maupun virtual. Peretasan pada keamanan jaringan komputer seringkali dilakukan oleh sekelompok orang yang ingin menembus suatu keamanan sebuah sistem. Aktivitas ini bertujuan untuk mencari, mendapatkan, mengubah, dan bahkan menghapus informasi. Maka dari itu penggunaan OpenvSwitch serta penerapan NIDS di dalamnya sangat berguna untuk keamanan jaringan virtual. Pada penelitian ini dilakukan simulasi untuk mencari tau seberapa besar persentase keberhasilan NIDS dalam mendeteksi serangan DOS yang diarahkan ke OpenvSwitch. Dari hasil penelitian ini, Persentase keberhasilan Snort dalam mendeteksi serangan di pengujian ini yaitu sekitar ± 75%.

Corresponding Author. Email: 672018172@student.uksw.edu [1].

1. Latar Belakang

Dalam lingkungan Server tervirtualisasi, biasanya diperlukan penggunaan *vSwitch* (*Virtual Switch*) agar dapat meneruskan lalu lintas antara *Virtual Macine* (VM) yang tersedia pada Host fisik yang sama antara VM serta jaringan fisik [4]. *OpenvSwitch* adalah perangkat lunak *Switch Open Source* yang dirancang untuk digunakan sebagai *vSwitch* di lingkungan virtual. *OpenvSwitch* terbuka untuk ekstensi dan kontrol terprogram menggunakan *OpenFlow* (OF-SPEC) dan manajemen protokol OVSDB (*Open vSwitch Database*) [7].

Keamanan pada Jaringan Komputer sangat penting untuk menjaga validitas serta integritas pada data dan menjamin ketersediaan layanan bagi penggunanya [6]. Agar sistem Jaringan Komputer tidak terganggu atau diserang oleh Peretas (*Hacker*), maka diperlukan sistem keamanan jaringan yang dapat mengamankan serta mencegah serangan dari Peretas (*Hacker*) tersebut. Keamanan jaringan menurut Mariusz Stawowski (2007) dalam jurnalnya yang berjudul "*The Principles of Network Security Design*" mengatakan bahwa keamanan jaringan yang utama sebagai perlindungan bagi sumber daya sistem terhadap ancaman yang berasal dari luar jaringan.

Banyak jenis serangan yang mampu dibuat oleh peretas agar bisa masuk ke Sistem Komputer yang dituju [1]. Salah satu serangan yang sering dilakukan oleh para peretas yaitu serangan DoS (*Denial of Service*). DoS merupakan salah satu metode serangan *Cyber* paling populer dalam keamanan jaringan. Menurut *Kaspersky Lab* dan *B2B International*, lebih dari 40% bisnis di dunia telah menjadi korban dari serangan DoS [1]. DoS merupakan serangan yang memiliki tujuan untuk mempengaruhi trafik jaringan sehingga jaringan tersebut tidak dapat digunakan oleh pengguna yang berhak atau sah. Serangan DoS dilakukan dengan cara membanjiri *Ip Address* jaringan target dengan Request sehingga sistem menjadi *Crash* atau kinerja dari pernagkat turun karena beban CPU yang tinggi. Serangan DoS memiliki berbagai macam tipe yaitu *Ping of Death*, *Syn Attack*, *Land Attack*, *UDP Flood*, dan *Smurf Attack* [1]. Maka dari itu diperlukan sistem keamanan pada Jaringan Komputer dengan melengkapi keamanan Jaringan Komputer menggunakan sistem yang dapat mendeteksi adanya serangan atau Intrusi (*Intrusion*). Sistem tersebut dikenal dengan istilah Sistem Deteksi Intrusi atau *Intrusion Detection System* (IDS).

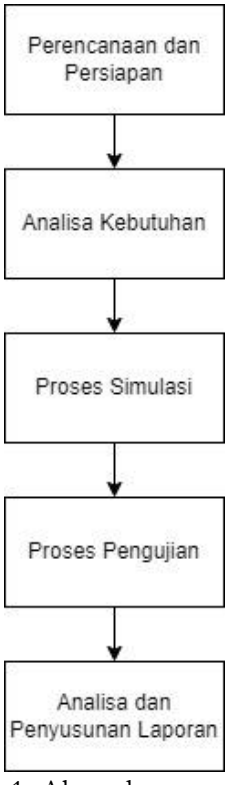
Penelitian terdahulu yang menjadi inti dari penelitian ini dilakukan oleh (Fachri & Harahap, 2020) dalam peneilitiannya yang berjudul "Simulasi Penggunaan Intrusion Detection System (IDS) Sebagai Keamanan Jaringan dan Komputer ", keamanan jaringan komputer merupakan bagian sebauh sistem yang sangat penting utnuk menjaga validitas dan integritas data serta menjamin ketersediaan layanan bagi pengguna dari mana dan kapan saja. Dan disatu sisi manusia sudah sangat tergantung dengan sistem informasi. Hal itu menyebabkan statistik insiden keamanan jaringan terus meningkat tajam dari tahun ke tahun. Salah satu cara yang dapat digunakan untuk menanggulangi atau mengatasi hal tersebut adalah dengan menggunakan *Intrusion Detection System* (IDS). Salah satu aplikasi yang mendukung intrusion detection system (IDS) adalah *Snort*. *Snort* mampu melakukan analisis terhadap bentuk serangan *intruder* yang menyalahgunakan protokol jaringan [5].

Lalu yang kedua dilakukan oleh (Purba & Efendi, 2020) pada penelitiannya yang berjudul "Perancangan dan Analisis Sistem Keamanan Jaringan Komputer Menggunakan SNORT "menjelasakan bahwa dengan menggunakan *Firewall* saja sistem kemanan jaringan tidak akan terjamin keamanannya. Maka diperlukan sebuah sistem untuk menjaga keamanan jaringan tersebut, yaitu *SNORT*. *SNORT* merupakan perangkat lunak yang akan memberikan peringatan ketika terjadi penyusupan kedalam sistem komputer [3].

Dan yang terakhir dilakukan oleh Iryani *dkk* (2021) pada penelitiannya yang berjudul "Analisis Performasi Data Plane Development Kit Terhadap Open Virtual Switch pada Jaringan Virtual", mengatakan virtualisasi merupakan teknologi yang digunakan untuk merubah sesuatu yang bersifat fisik ke dalam bentuk virtual. Jaringan virtual dalam virtualisasi membutuhkan *switch* virtual untuk menghubungkan sejumlah virtual mesin. *Switch* virtual yang umum digunakan adalah *Open vSwitch*. Paket yang melewati *Open vSwitch* diteruskan dengan cara mencocokkan *Flow Entries* yang ada di *Flow Table*. Hal ini membuat pemrosesan paket relatif lebih lambat karena melewati banyak *Space* [2].

Dalam penelitian ini, penulis melakukan sebuah simulasi keamanan jaringan *virtual* dengan menggunakan jaringan yang dibuat melalui *OpenvSwitch*. Pada *OpenvSwitch* tersebut akan ditambahkan modul baru yaitu DPDK (*Data Plane Development Kit*). Fungsi dari DPDK ini ialah menambah kinerja dalam memproses paket dengan lebih efisien. lalu dilakukan penerapan sistem kemanan jaringan pada jaringan virtual tersebut menggunakan *Network-Based Intrusion Detection System* (NIDS). Keamanan dari jaringan virtual tersebut akan di uji coba dengan memberikan serangan DOS layer 2 berupa *Ping of Death* (PoD) dan *Syn Flooding*, untuk melihat seberapa besar persentase keberhasilan sistem NIDS yang diterapkan di OVS.

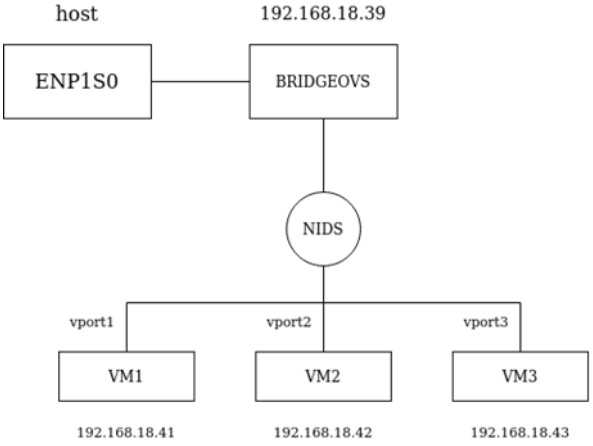
2. Metode Penelitian



Gambar 1. Alur tahapan penelitian

Pada Gambar 1 menggambarkan alur dari metode penelitian ini. Pada tahap Perencanaan dan Persiapan dilakukan perencanaan skenario berdasarkan jurnal-jurnal serta referensi untuk simulasi yang akan dibuat dan melakukan persiapan *Hardware* dan *Software* guna melakukan simulasi sistem keamanan jaringan menggunakan NIDS *Snort* pada *OpenvSwitch*. Selanjutnya Melakukan analisa kebutuhan seperti sistem operasi yang akan digunakan, tipe IDS yang akan digunakan, versi *OpenvSwitch* dan DPDK yang digunakan, serta menentukan kebutuhan simulasi lainnya. Sistem operasi yang digunakan untuk melakukan penelitian ini adalah sistem operasi *Linux* dan menggunakan *Network-based Intrusion Detection System* (NIDS). Pada Proses Simulasi memiliki beberapa kegiatan seperti perancangan jaringan, instalasi perngakat lunak, konfigurasi *OpenvSwitch* dan konfigurasi NIDS. Selanjutnya dilakukan Proses Pengujian. Hasil dari pengujian yang dilakukan yaitu komputer yang bekerja sebagai penyerang (*Attacker*) akan melakukan beberapa serangan terhadap komputer atau Server yang berada dalam jaringan lalu komputer yang telah dilakukan instalasi NIDS mendeteksi serangan apa saja yang dilakukan oleh komputer penyerang. Setelah proses pengujian selesai maka tahap selanjutnya yaitu membuat analisa dan penyusunan laporan berdasarkan hasil pengujian simulasi.

Topologi jaringan yang akan digunakan pada penelitian ini dapat dilihat pada Gambar 2. Host merupakan perangkat yang menjalankan OVS. Pada OVS akan diterapkan sistem keamanan jaringan yaitu *Network Intrusion Detection System* (NIDS) dan membuat jaringan virtual yang terdiri dari 3 buah VM, yaitu VM 1, VM 2, dan VM 3 dengan menggunakan VM *VirtualBox*. Tiap VM dihubungkan ke *Bridge bridgeovs* pada OVS dengan Virtual Interface (*vport1–vport3*). Komputer yang digunakan memiliki satu *Pyhsical Interface* yaitu *enp1s0*. *enp1s0* berfungsi untuk menghubungkan *Host* dengan OVS, VM, dan internet.

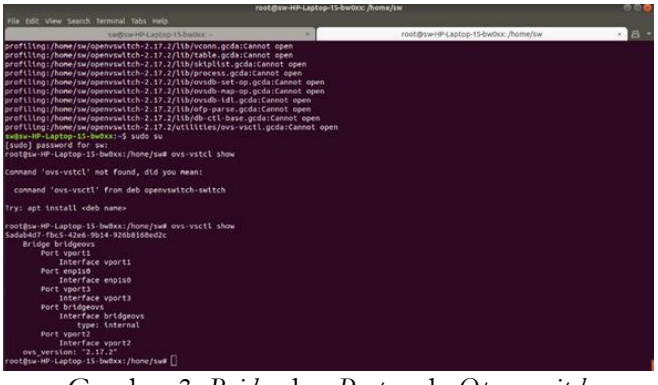


Gambar 2. Topologi jaringan pada penelitian.

3. Hasil dan Pembahasan

Hasil yang akan didapatkan dari penelitian ini ialah untuk mengetahui seberapa banyak serangan yang berhasil dideteksi oleh NIDS yang diarahkan ke *OpenvSwitch*. Setelah persiapan yang dimana topologi, perangkat dan *Software* seperti DPDK, *OpenvSwitch*, *Snort*, serta *Tool Ping of Death* dan *Syn Flooding* yang diunduh pada VM Attacker untuk melakukan serangan terhadap *OpenvSwitch* telah dipersiapkan, maka langkah selanjutnya yaitu melakukan simulasi penyerangan berupa serangan *Ping of Death* dan *Syn Flooding* kepada *OpenvSwitch* dan melihat apakah sistem NIDS dari *Snort* dapat mendeteksi serangan tersebut.

Sebelum melakukan instalasi pada *OpenvSwitch*, hal pertama yang dilakukan yaitu melakukan penginstalan DPDK. DPDK dapat diunduh melalui *https://fast.dpdk.org/rel/* dan *OpenvSwitch* dapat di unduh melalui *https://www.openvswitch.org/download/*. Agar *OpenvSwitch* dapat menggunakan DPDK dan *Libraries*-nya, pada perintah untuk melakukan konfigurasi paket dari *OpenvSwitch* ditambahkan *--with-dpdk=yes*. Maka perintah untuk konfigurasi paketnya menjadi *./configure --with-dpdk=yes*. Setelah melakukan instalasi *OpenvSwitch*, hal pertama yang dilakukan yaitu membuat *Bridge* dan 3 Port untuk menyambungkan ketiga VM agar mendapatkan jaringan. Pada penelitian ini *Bridge* diberi nama *bridgeovs* lalu untuk ketiga port diberi nama *vport1*, *vport2*, dan *vport3*.



Gambar 3. *Bridge* dan *Port* pada *Openvswitch*

Bridge dan ketiga VM tersebut memiliki *Ip address* sebagai berikut :

1) *bridgeovs* : 192.168.18.39
2) VM1 yang terhubung melalui *vport1* : 192.168.18.41
3) VM2 yang terhubung melalui *vport2* : 192.168.18.42
4) VM3 yang terhubung melalui *vport3* : 192.168.18.43

Setelah DPDK dan *OpenvSwitch* berhasil diinstal selanjutnya dilakukan penginstalan *Snort* yang merupakan sebuah *Tools* untuk menerapkan NIDS. Sebelum melakukan penginstalan dan konfigurasi *Snort*, Agar *Snort* dapat menggunakan DPDK diperlukan modul DPDK DAQ pada *Snort* agar NIDS dapat berjalan pada DPDK. DPDK DAQ dapat diinstal pada *https://github.com/napatech/daq_dpdk_multiqueue*. untuk melakukan konfigurasi dari DPDK DAQ ini dapat dilakukan dengan memasukan perintah :
./ configure --with-dpdk-includes=<dpdk-16.07 dir>/x86_64-native-linuxapp-gcc/include --with-dpdk-libraries=<dpdk-16.07 dir>/x86_64-native-linuxapp-gcc/lib

Setelah DPDK DAQ diunduh dan dikonfigurasi, selanjutnya *Snort* dapat diunduh. Setelah selesai mengunduh *Snort*, terlebih dahulu untuk melakukan konfigurasi agar *Snort* dapat menangkap dan membaca paket-paket yang ditujukan kepada *OpenvSwitch*. Konfigurasi yang dilakukan yaitu seperti konfigurasi pada *Networks cards* untuk memberikan alamat IP pada *Snort* agar *Snort* dapat mengetahui alamat IP yang manakah yang harus di pantau dan membuat *Rules* pada direktori *Snort*. Penambahan *Rules* pada *Snort* sangat penting agar *Snort* bisa membedakan paket-paket yang menuju *Openvswitch*, apakah paket yang dikirim merupakan paket yang berbahaya atau tidak. *Rules* yang perlu ditambahkan yaitu :

1) *alert icmp any any -> $HOME_NET any (msg:"Ping of Death"; dsize:>1500; sid:3000003; rev:1;)* (untuk mengetahui jika ada serangan Ping of Death) ,
2) *alert tcp any any -% $HOME_NET any (msg:"SJSU EE-209 – SYN Flood attack"; flow: stateless flags:S; detection_filter: track by_dst, count 500, seconds 5, sid: 293;)* (untuk mengetahui jika ada serangan Syn Flooding) dan,
3) *alert icmp any any -> any any (msg:"ICMP Traffic Detected"; sid:10000001; metadata:policy security-ips alert;)* (untuk mengetahui jika ada yang melakukan Ping).

Setelah konfigurasi telah selesai, maka *Snort* perlu diaktifkan agar dapat membaca paket-paket yang sedang menuju ke *OpenvSwitch*. Untuk mengaktifkan *Snort,* perlu memasukan perintah pada terminal yaitu perintah */usr/local/bin/snort -v*. Agar *Snort* dapat menjalankan sistem NIDS, maka perlu dimasukan perintah pada terminal yaitu :

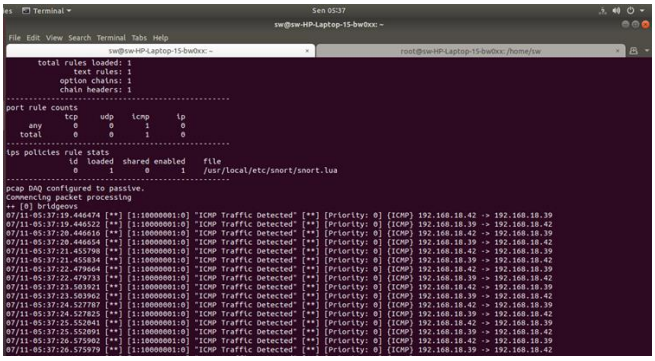
*sudo snort-c usr/local/etc/snort/snort.lua -R usrlocal/etc/rules/local.rules *
-i brifdgeovs -A alert_fast -s 65535 -k none

Agar *Snort* dapat dijalankan menggunakan DPDK, *Snort* dijalankan menggunakan *Passive mode* dengan DPDK DAQ dengan memasukkan perintah :

taskset -c 0-13 snort --daq dpdk --daq-var dpdk_argc="-n4" -i "dpdk0 dpdk1" -z 14 --daq-var dpdk_queues=14

Snort memperoleh paket melalui DPDK DAQ dan menerapkan pemrosesan paket internal ke setiap paket. Pada proses akhirnya, DPDK DAQ dapat mengetahui keputusan yang diambil oleh mesin untuk setiap paket seperti paket yang lolos, paket yang terblok, dll. jika paket diblokir maka paket akan diteruskan. Terkadang *Snort* dapat mengirimkan kembali apabila paket melewati *Backward Path*, seperti *Injected Packets* yang dibuat dan dikirim oleh *Snort* itu sendiri.

Selanjutnya dilakukan pengujian *Ping* terlebih dulu Agar dapat mengetahui apakah *Snort* dapat mendeteksi *Ping* dari VM, maka dilakukan *Ping* dari VM ke *OpenvSwitch* dengan memasukan perintah *ping 192.168.18.39*. Rata–rata total waktu yang dibutuhkan agar paket dapat terkirim ke *OpenvSwitch* dari ketiga VM tersebut yaitu sekitar <1 *Milisecond* (ms) dan juga TTL (*Time to live*) yang merupakan waktu maksimum *OpenvSwitch* untuk menerima paket yang dikirimkan oleh ketiga VM tersebut yaitu dengan jumlah 64.

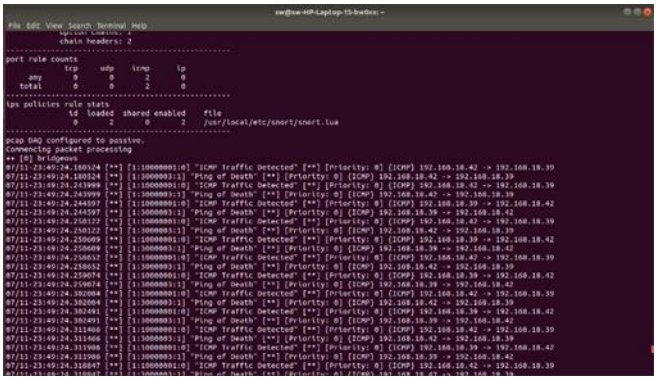


Gambar 4. Hasil *Ping* yang terdeteksi oleh *Snort*

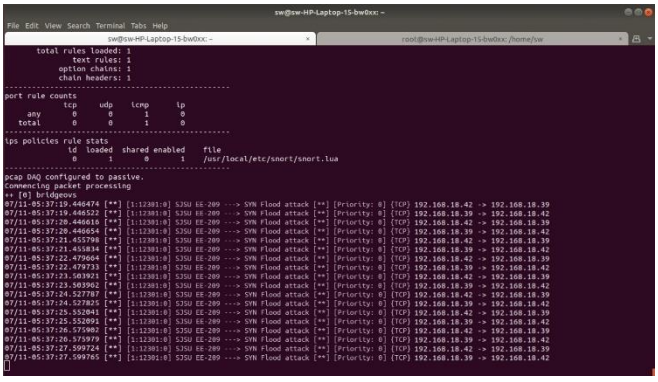
Pada Gambar 4 dapat dilihat bahwa *Snort* mendeteksi adanya *Ping* menuju *OpenvSwitch*. *Snort* juga menunjukan waktu dan tanggal serta alamat IP yang mencoba melakukan Ping ke *OpenvSwitch*.

Setelah pengujian *Ping* berhasil dilakukan, selanjutnya dilakukan simulasi penyerangan dari *Virtual Machine* ke *OpenvSwitch*. VM 2 yang merupakan *Attacker* akan melakukan serangan *Ping of Death* dan *Syn Flooding* ke *OpenvSwitch*. Pada percobaan ini Attacker memiliki alamat IP 192.168.18.42 yang akan melakukan *Ping of Death* dan *Syn Flooding* ke *OpenvSwitch* dengan alamat IP 192.168.18.39. Perintah untuk melakukan *Ping of Death* yaitu *pod 192.168.18.39*. Lalu untuk serangan *Syn Flooding* dijalankan dengan menggunakan perintah *sudo hping3 –S --flood 192.168.18.39*.

Pada gambar 5 dapat dilihat bahwa serangan *Ping of Death* dari VM 2 telah dideteksi oleh *Snort*. *Snort* akan menampilkan pesan mengenai serangan " *Ping of Death* " yang ditujukan kepada *OpenvSwitch* serta menampilkan waktu penyerangan, *Port* dan alamat IP mana yang menyerang *OpenvSwitch*. Begitu pula dengan serangan *Syn Flooding* yang dapat dilihat pada gambar 6. *Snort* berhasil mendeteksi serangan *Syn Flooding* dan *Snort* menampilkan pesan mengenai serangan " *SYN Flood attack* ".



Gambar 5. *Snort* yang mendeteksi adanya serangan *Ping of Death*



Gambar 6. *Snort* yang mendeteksi adanya serangan *Syn Flooding*

Snort mendeteksi serangan *Ping of Death* dan *Syn Flooding* yang dilancarkan oleh *Attacker* dan memperlihatkan waktu, *Port*, serta alamat IP dari penyerang. Dalam pengujian ini dilakukan 20 kali percobaan serangan dan 10 kali *Ping* terhadap *Switch*. Pada percobaan tersebut *Snort* menghasilkan 19 kali *False Positive* dan 11 kali *False Negative*. *False Positive* yang dideteksi oleh *Snort* yaitu serangan *Ping of Death* sebanyak 7 kali, serangan *Syn Flooding* sebanyak 6 kali dan *Ping* sebanyak 7 kali. Pada serangan yang tidak terdeteksi *Snort* hanya mendeteksi Ping biasa pada serangan tersebut. *Ping* yang mengalami *False Negative* dideteksi oleh Snort sebagai serangan *Ping of Death*.

Tabel 1. Hasil percobaan serangan

Percobaan	*Ping of Death*	*Syn Flooding*	*Ping*
1	Berhasil	Berhasil	Berhasil
2	Berhasil	Gagal	Gagal
3	Gagal	Berhasil	Berhasil
4	Berhasil	Gagal	Berhasil
5	Berhasil	Berhasil	Gagal
6	Berhasil	Gagal	Berhasil
7	Gagal	Berhasil	Berhasil
8	Berhasil	Gagal	Berhasil
9	Gagal	Gagal	Berhasil
10	Berhasil	Berhasil	Gagal

4. Kesimpulan

Hasil dari pengujian menunjukan bahwa DPDK *OpenvSwitch* dapat memberikan akses internet dari jaringan lokal pada *Virtual Machine* yang terhubung dengan baik di *OpenvSwitch* dan juga implementasi sistem keamanan jaringan *Network-Intrusion Detection System* (NIDS) menggunakan *Snort* dalam DPDK *OpenvSwitch* dapat mendeteksi serangan – serangan yang telah dilancarkan. Persentase keberhasilan *Snort* dalam mendeteksi serangan di pengujian ini yaitu: *Ping of Death* sebesar 70%, *Syn Flooding* 60%, dan *Ping* 70% dengan rata – rata dari 30 serangan yang dilancarkan *Snort* dapat mendeteksi serangan sebanyak ± 75%. Dengan jumlah persentase tersebut, penggunaan *Snort* yang dibantu dengan DPDK sebagai NIDS yang diterapkan di DPDK *OpenvSwitch* dapat berjalan dengan baik dan mampu mendeteksi seranga-serangan yang dilancarkan walaupun hasilnya tidak mencapai 100%.

5. Daftar Pustaka

[1] Antony, F. and Gustriansyah, R., 2021. Deteksi Serangan Denial of Service pada Internet of Things Menggunakan Finite-State Automata. *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, *21*(1), pp.43-52. DOI: https://doi.org/10.30812/matrik.v21i1.1078.

[2] Iryani, N., Ramadhani, A.D. and Ramadhani, Q.P. 2021. Analisis Performansi Data Plane Development Kit Terhadap Open Virtual Switch pada Jaringan Virtual, JTERA (Jurnal Teknologi Rekayasa), 6(1), p. 93. DOI: https://doi.org/10.31544/jtera.v6.i1.2021.93-100.

[3] Purba, W.W. and Efendi, R., 2020. Perancangan dan analisis sistem keamanan jaringan komputer menggunakan SNORT. *AITI*, *17*(2), pp.143-158. DOI: https://doi.org/10.24246/aiti.v17i2.143-158.

[4] Lowe, W., 2010. *VMware Infrastructure 3 for Dummies*. John Wiley & Sons.

[5] Fachri, B. and Harahap, F.H., 2020. Simulasi Penggunaan Intrusion Detection System (IDS) Sebagai Keamanan Jaringan dan Komputer. *Jurnal Media Informatika Budidarma*, *4*(2), pp.413-420. DOI: https://doi.org/10.30865/mib.v4i2.2037.

[6] Rozi, F., 2021. *ANALISIS SISTEM KEAMANAN JARINGAN MENGGUNAKAN METODEINTRUSION DETECTION SYSTEM (IDS), INTRUSION PREVENTION SYSTEM (IPS), DAN HONEYPOT* (Skripsi, Universitas Yudharta).

[7] OpenvSwitch. 2022. Open vSwitch, Release 2.17.90.

[8] Dietrich, N. 2021. Snort 3.1.18.0 on Ubuntu 18 & 20 " Configuring a Full NIDS & SIEM.

[9] ANam, M.K., Sudyana, D., Ulfah, A.N. and Lizarti, N., 2020. Optimalisasi Penggunaan VirtualBox Sebagai Virtual Computer Laboratory untuk Simulasi Jaringan dan Praktikum pada SMK Taruna Mandiri Pekanbaru. *J-PEMAS-Jurnal Pengabdian Masyarakat*, *1*(2), pp.39-44.

[10] Suwanto, R., Ruslianto, I. and Diponegoro, M., 2019. Implementasi intrusion prevention system (IPS) menggunakan snort dan IPTABLE pada monitoring jaringan lokal berbasis website. *Coding Jurnal Komputer dan Aplikasi*, *7*(01). DOI: http://dx.doi.org/10.26418/coding.v7i01.32690.

[11] Walad, I., 2020. *Analisis Denial Of Service Attack Pada Sistem Keamanan Web* (Skripsi, Universitas Sumatera Utara). Available at: http://repositori.usu.ac.id/handle/123456789/28240.

[12] Sun, G., Li, W. and Wang, D., 2018. Performance Evaluation of DPDK Open vSwitch with Parallelization Feature on Multi-Core Platform. *J. Commun.*, *13*(November), pp.685-690. DOI: https://doi.org/10.12720/jcm.13.11.685-690.

[13] Fanani, G. and Riadi, I., 2020. Analysis of Digital Evidence on Denial of Service (DoS) Attack Log Based. *Buletin Ilmiah Sarjana Teknik Elektro*, *2*(2), pp.70-74. DOI: https://doi.org/10.12928/biste.v2i2.1065.

[14] Munawar, Z. and Putri, N.I., 2020. Keamanan Jaringan Komputer Pada Era Big Data. *J-SIKA| Jurnal Sistem Informasi Karya Anak Bangsa*, *2*(01), pp.14-20.

[15] Hafiz, A., Kurniawan, T., Sivi, N.A., Ikhsan, F.K. and Andhika, P., 2020. Analisis Celah Keamanan Jaringan Dan Server Menggunakan Snort Intrusion Detection System. *Jurnal Informasi Dan Komputer*, *8*(2), pp.55-65. DOI: https://doi.org/10.35959/jik.v8i2.185.