Jurnal Teknik Informatika (JUTIF)
Vol. 6, No. 1, February 2025, pp. 173-186
DOI: https://doi.org/10.52436/1.jutif.2025.6.1.1830
p-ISSN: 2723-3863
e-ISSN: 2723-3871

MODELING INTRUSION DETECTION AND PREVENTION SYSTEM TO DETECT AND PREVENT NETWORK ATTACKS USING WAZUH

Otniel Dewangga Divan Pramudya[1], Puspanda Hatta*[2], Cucuk Wawan Budiyanto[3]

[1,2,3]Informatics and Computer Engineering Education, Faculty of Teacher Training and Education, Universitas Sebelas Maret, Indonesia
Email:[2]hatta.puspanda@staff.uns.ac.id

(Article received: February 3, 2023; Revision: October 24, 2024; published: February 20, 2025)

Abstract

The rapid development of technology has a positive impact on society. The internet can be easily accessed anytime and anywhere, but with the advancement of internet technology, there are many threats lurking in the security of its users. Criminal activities in the digital world are referred to as cybercrime. Numerous cases of cybercrime have occurred worldwide, ranging from attacks that can disable servers to data theft and illegal access. It is noted that more than 50% of companies do not have a plan to respond to these cybercrimes. This is due to various factors, one of which is the limited availability of freely accessible and easily configurable network security platforms for all users. Therefore, this research aims to provide a solution in the form of an open-source-based Intrusion Detection and Prevention System (IDPS) that can be freely distributed and easily configured, one of which is Wazuh. The study uses the Cisco PPDIOO approach in developing a virtual lab with various scenarios for testing and measuring the Quality of Services (QoS) of Wazuh's performance. From the created test scenarios, Wazuh can detect attacks from both inside and outside the network. Wazuh has proven to be capable of detecting and preventing various types of network attacks and features that can facilitate users in responding to cybercrime, making it a potential solution for organizations that have not planned to respond to cybercrime.

Keywords: *Cybercrime, IDPS, Open-Source, QoS, Virtual Lab, Wazuh*

PEMODELAN *INTRUSION DETECTION AND PREVENTION SYSTEM* UNTUK MENDETEKSI DAN MENCEGAH SERANGAN JARINGAN MENGGUNAKAN WAZUH

Abstrak

Perkembangan teknologi yang pesat memberikan dampak positif bagi masyarakat. internet dapat dengan mudah diakses kapanpun dan dimanapun, namun dengan berkembangnya teknologi internet, ada banyak ancaman yang mengintai keamanan dari para penggunanya. Tindak kejahatan dalam dunia digital ini disebut dengan *cybercrime.* Sudah banyak kasus *cybercrime* yang terjadi dari seluruh penjuru dunia, mulai dari serangan yang dapat melumpuhkan *server*, hingga pencurian data dan akses ilegal. Tercatat ada lebih dari 50% perusahaan yang belum memiliki rencana untuk merespons *cybercrime* ini. Hal ini diakibatkan oleh banyak faktor, salah satunya adalah terbatasnya platform keamanan jaringan yang bebas digunakan oleh semua kalangan dan mudah dikonfigurasi. Oleh karena itu, penelitian ini bertujuan untuk memberikan solusi berupa IDPS berbasis *open-source* yang dapat disebarluaskan secara bebas dan mudah dikonfigurasi, salah satunya adalah Wazuh. Penelitian ini menggunakan metode pendekatan Cisco PPDIOO dalam mengembangkan sebuah lab virtual yang memiliki beberapa skenario untuk pengujian serta pengukuran *Quality of Services* (QoS) dari kinerja Wazuh. Dari beberapa skenario yang telah dibuat untuk pengujian, Wazuh dapat mendeteksi serangan dari dalam maupun luar *network*. Wazuh terbukti dapat mendeteksi dan mencegah berbagai macam serangan jaringan dan memiliki fitur yang dapat memudahkan pengguna dalam merespons *cybercrime* dan dapat menjadi solusi untuk organisasi yang belum berencana merespons *cybercrime.*

Kata kunci: *Cybercrime, IDPS, Lab Virtual, Open-source, QoS, Wazuh*

1. PENDAHULUAN

Perkembangan teknologi yang pesat saat ini telah mengubah bagaimana cara kita menjalani

kehidupan sehari-hari. Kegiatan seperti berbagi informasi, berkomunikasi, berbelanja, dan mencari hiburan kini dapat dilakukan secara instan dengan layanan internet. Hal ini sangat berbeda jika dibandingkan dengan beberapa dekade lalu dimana kegiatan di atas masih dilakukan dengan cara tradisional.

Dengan berkembangnya teknologi dan layanan internet, server berperan sangat penting dalam menyediakan berbagai macam informasi bagi pengguna internet. Server adalah sebuah sistem komputer yang menyediakan berkas, database, email dan layanan lain yang dapat diakses oleh client yang berada dalam jaringan yang sama [1]. Server memiliki berbagai macam jenis sesuai dengan fungsinya, seperti File Server untuk menyimpan berkas, Web Server untuk menyimpan dan menampilkan Website, App Server untuk menyimpan dan menampilkan aplikasi, dan masih banyak lagi.

Namun, dengan berkembangnya teknologi serta internet juga membuat ancaman yang mengintai ikut beradaptasi untuk mencari celah keamanan dan mendapatkan akses ilegal ke dalam data pengguna. Kejahatan yang terjadi di dalam dunia digital biasa disebut dengan *cybercrime.* Menurut Raharjo dkk. [2], *cybercrime* atau kejahatan siber adalah sebuah tindakan terlarang oleh peraturan dan hukum yang melibatkan penggunaan teknologi digital dan digunakan untuk menyerang teknologi itu sendiri.

Pada tahun 2020, McAfee melaporkan adanya kerugian global mencapai US$1 triliun yang disebabkan oleh *cybercrime* [3]. Serangan ini meningkat sebanyak 50% dari tahun 2018. Kerugian yang disebabkan *cybercrime* bukan hanya berupa biaya, namun juga reputasi dan efisiensi dari perusahaan yang menurun. McAfee juga mencatat bahwa 56% dari perusahaan yang disurvey tidak memiliki rencana dalam mencegah maupun merespon *cybercrime*.

Di Indonesia, *cybercrime* juga sering terjadi dan menimbulkan banyak kerugian. Portal berita liputan6.com [4] mengabarkan bahwa pada tahun 2020 lalu telah terjadi serangan *Distributed Denial of Service* (DDoS) pada situs corona.jakarta.go.id yang merupakan situs untuk memantau persebaran virus *Corona* di daerah DKI Jakarta. Serangan ini menyebabkan kerugian bagi pengguna serta pengelola, yang mana pengguna dirugikan karena tidak dapat mengakses situs sementara waktu, dan pengelola mendapatkan reputasi yang buruk karena situs yang dikelolanya ternyata tidak aman.

Selain itu, situs kompas.com [5] memberitakan bahwasanya telah terjadi peretasan terhadap BPJS Kesehatan pada Mei 2021 dimana data dari 279 juta warga Indonesia telah diperjualbelikan pada forum online bernama “Raid Forums” oleh akun bernama “Kotz”. Data yang berisi nomor kartu, NIK, nomor ponsel, dan lain-lain ini diperjualbelikan seharga 0,15 *Bitcoin* atau setara dengan Rp 81,6 juta pada saat berita ditulis. Hal ini tentu membuat masyarakat meragukan keamanan jaringan yang dimiliki oleh pengelola BPJS Kesehatan.

Beberapa peristiwa di atas menunjukkan kerentanan sebuah *server* terhadap serangan bahkan jika yang mengelola adalah instansi pemerintahan sekalipun. Melihat data dari McAfee sebelumnya, lebih dari setengah jumlah perusahaan yang disurvey tidak memiliki rencana untuk mengantisipasi bahkan mendeteksi serangan *cybercrime.* Padahal, ada banyak cara untuk mengantisipasi dan mencegah serangan, salah satunya dalah menggunakan *Intrusion Detection and Prevention System* (IDPS). IDPS berguna untuk mendeteksi serta mencegah serangan yang terjadi dalam jaringan, baik itu potensi serangan *malware* / virus, hingga aktivitas mencurigakan dalam jaringan tersebut. IDPS bekerja dengan cara mendeteksi serangan sesuai dengan konfigurasi yang sudah diterapkan, lalu mengirimkan *alert* yang biasanya berupa *log* kepada pengguna sehingga pengguna dapat menindaklanjuti secara manual maupun otomatis dengan cara menerapkan konfigurasi yang sudah dibuat [6].

Tetapi semakin lengkap fitur dari sebuah IDPS, *cost* yang diperlukan juga semakin tinggi [7], oleh karena itu diperlukan sebuah IDPS yang ramah bagi organisasi maupun perusahaan yang sedang merintis. IDPS *open-source* bisa menjadi jawaban bagi permasalahan ini dikarenakan *software open-source* adalah *software* yang lisensinya tidak dimiliki oleh individu maupun organisasi tertentu [8] sehingga bebas digunakan, dimodifikasi, dan dibagikan oleh dan kepada siapapun.

Keamanan jaringan menurut Kumar [9] adalah pengawasan dan peraturan yang ditawarkan penanggungjawab sebagai upaya untuk mendapatkan perlindungan dari akses tidak sah, penyalahgunaan serta modifikasi yang tidak diinginkan yang dapat dilakukan melalui jaringan. Keamanan jaringan mendukung serta menjamin aspek-aspek keamanan informasi [10] seperti kerahasiaan (*Confidentiality*), integritas (*Integrity*), dan ketersediaan (*Availability*).

Ada banyak ancaman yang mengintai keamanan jaringan, seperti *Malware* yang merupakan segala macam bentuk program berbahaya seperti virus, *spyware*, *trojan*, *adware*, dan kode-kode lain [11] yang diluncurkan oleh pelaku peretasan untuk mendapat akses ilegal terhadap perangkat korban [12]. Kemudian ada serangan *DoS* yang merupakan serangan yang berusaha untuk membuat perangkat target terbebani dengan permintaan dari penyerang sehingga terjadi *flooding* yang menyebabkan penolakan layanan dari target [13].

Beberapa jenis serangan DoS diantaranya seperti *UDP Floods, ICMP Floods, SYN Floods* dan *Slowloris* [14]. Selanjutnya ada serangan *SQL Injection* yang dapat membuat penyerang memiliki

akses penuh terhadap database target dengan cara mengeksekusi *query* melalui celah yang ada, hal ini membuat penyerang dapat melakukan apa saja yang diinginkan seperti pencurian data, menghapus data, dan lain-lain [15].

Kemudian ada *Bruteforce Attack* yang merupakan sebuah jenis serangan dimana penyerang akan mencoba banyak hingga semua kemungkinan yang dapat membuka akses kepada perangkat target. Serangan ini membutuhkan waktu yang relatif lama namun jika dibiarkan maka mungkin saja penyerang akhirnya mendapatkan akses ke perangkat target [16]. Beberapa ancaman yang telah disebutkan hanyalah sedikit dari banyaknya ancaman yang mengintai keamanan jaringan, maka dari itu diperlukan sebuah platform keamanan jaringan yang dapat mengantisipasinya.

Wazuh adalah sebuah platform IDPS *open-source* yang dapat mendeteksi, mencegah dan merepons berbagai ancaman jaringan. Wazuh dapat digunakan pada lingkungan *on-premise,* virtual, *container* dan juga *cloud* [17]. Wazuh memiliki beberapa komponen [18] untuk dapat bekerja, seperti Wazuh Indexer yang merupakan mesin utama dari server Wazuh, Wazuh Server / Manager yang bertugas untuk menganalisis data yang diterima dari Wazuh Agent, Wazuh Dashboard yang merupakan tampilan antarmuka web yang ramah pengguna, dan Wazuh Agent yang terpasang pada *endpoint* yang ingin diamati seperti laptop, desktop, maupun server.

Beberapa penelitian telah dilakukan dalam menggunakan IDPS sebagai upaya untuk mendeteksi dan mencegah serangan jaringan, diantaranya penelitian yang dilakukan oleh Ernawati dan Rachmat [19] yang merancang dan mengimplementasikan sistem keamanan jaringan menggunakan *cowrie honeypot* dan *snort-inline mode* sebagai IPS, dan menghasilkan kesimpulan bahwa *snort-inline mode* lebih direkomendasikan untuk digunakan karena lebih unggul dalam uji parameter. Selanjutnya penelitian oleh Harahap A. dan Hutrianto [20], menunjukkan kemampuan Wazuh sebagai IDS dalam mendeteksi dan mencegah serangan seperti SSH *Bruteforce,* DoS, dan lain-lain. Kemudian pada penelitian yang dilakukan oleh Husain et al. [21] menunjukkan implementasi keamanan server menggunakan IDPS yaitu snort. Snort sebagai IDPS dapat mencegah serangan jaringan seperti port scanning dan akses FTP. Selanjutnya penelitian oleh Yomo et al. [22] menggunakan Wazuh untuk mendeteksi serangan SQL *Injection*. Hasilnya adalah Wazuh dapat mendeteksi dan menampilkan 45,67% serangan yang dilakukan. Penelitian oleh Pratama et al. [23] juga menggunakan Wazuh untuk mendeteksi dan menampilkan serangan DoS. Hasil yang diperoleh adalah Wazuh dapat mendeteksi metode serangan SYN *flood* dan ICMP *flood*, namun tidak dengan ACK *flood* dan UDP *flood*.

Dari beberapa penelitian yang sudah dilakukan sebelumnya, belum ada yang mengimplementasikan Wazuh untuk mendeteksi dan mencegah serangan jaringan dalam beberapa skenario berbeda dan mengukur performanya. Maka dari itu, Penelitian ini bertujuan untuk menganalisis dan mengevaluasi kinerja IDPS dalam mendeteksi serta mencegah berbagai macam jenis serangan dalam beberapa skenario yang berbeda, dan juga untuk mengukur serta menganalisis kinerja IDPS tersebut menggunakan parameter QoS.

Penelitian ini akan menguji beberapa fitur Wazuh seperti deteksi perubahan (integritas) file, serangan bruteforce, port scanning, DoS, dan SQL Injection. Penelitian akan dilakukan di dalam Lab Virtual pada Host Windows 10, dengan Kali Linux sebagai penyerang, Ubuntu Server sebagai target serangan serta tempat dimana Wazuh akan dipasang, Ubuntu Desktop sebagai pengamat, Router Mikrotik, dan Swtich Cisco. Dari beberapa skenario yang akan dijalankan, performa yang akan diukur adalah selisih waktu respon (delay) antara target dan juga Wazuh.

Penelitian ini diharapkan dapat berkontribusi untuk menambah wawasan dalam upaya mendeteksi dan mencegah serangan jaringan baik skala besar maupun kecil. Juga diharapkan penelitian ini dapat menjadi referensi media pembelajaran dalam bidang keamanan jaringan, dan dapat diimplementasikan pada organisasi skala besar maupun kecil dalam upaya merespons *cybercrime.*

2. METODE PENELITIAN

2.1. Cisco PPDIOO

Penelitian ini akan dilakukan dengan menggunakan metode penelitian dari Cisco yaitu PPDIOO yang merupakan singkatan dari *Prepare, Plan, Design, Implement, Operate,* dan *Optimize.* Metode ini dapat memberikan keuntungan berupa *Total Cost of Ownership* (TCO) yang turun [24]. Adapun tahapan dari Cisco PPDIOO adalah sebagai berikut:

2.1.1. Prepare

Pada tahap ini, akan dilakukan persiapan dan analisis kebutuhan penelitian, diantara lain:

a. Koneksi internet
b. Microsoft Windows 10
c. VMWare Workstation Player 17
d. Image EVE-NG 5.0.1-19 Community Edition
e. Image Mikrotik CHR 6.49.10
f. Image Cisco Switch IOS 15.1
g. Image Ubuntu Server 22.04
h. Image Ubuntu Desktop 20.04
i. Image Kali Linux 2023.3

Windows 10 dipilih sebagai host karena meripakan sistem operasi yang paling banyak digunakan dalam 3 tahun terakhir [25] dan memiliki stabilitas yang baik dibandingkan dengan versi terbaru. VMWare Workstation dipilih karena dapat digunakan secara gratis dan memiliki fitur yang lengkap. EVE-NG dipilih untuk menjadi lab virtual

dikarenakan memiliki kompabilitas yang luas terhadap berbagai macam vendor, antarmuka yang ramah pengguna dan dapat secara gratis digunakan. Mikrotik CHR dan Cisco Switch dipilih karena dapat menggantikan peran router dan switch dalam sebuah lab virtual. Sistem operasi untuk vitual PC seperti Ubuntu Server, Ubuntu Desktop dan Kali Linux dipilih karena gratis dan juga stabil.

2.1.2. Plan

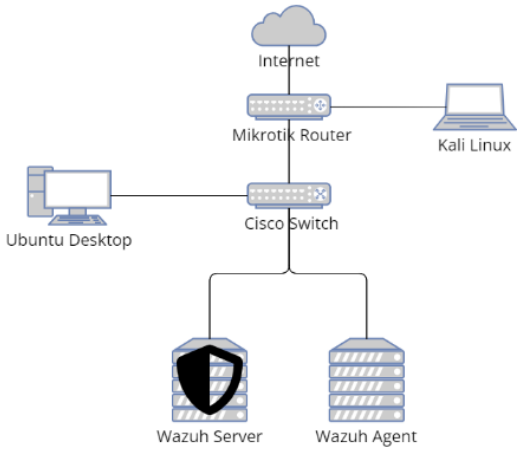
Setelah mempersiapkan kebutuhan untuk penelitian, selanjutnya disusun rencana agar tujuan dari penelitian tercapai. Adapun rencana dari penelitian ini adalah:

a. Microsoft Windows 10 beserta koneksi internet akan menjadi host dan dipasangi VMWare Workstation Player 17.
b. VMWare Workstation Player 17 akan dipasangi EVE-NG yang akan menjadi lab virtual dalam penelitian ini.
c. Lab virtual akan berisi *nodes* seperti Router Mikrotik, Switch Cisco, dan Virtual PC seperti Ubuntu Server, Ubuntu Desktop, dan Kali Linux.
d. Router Mikrotik dalam lab virtual akan mendapat koneksi internet dari Host dan akan dibagikan kepada *nodes* yang tersambung padanya.
e. Ubuntu Server berperan sebagai Wazuh Server yang akan menerima dan mengolah *alert* dan Wazuh Agent yang akan menjadi target penyerangan.
f. Ubuntu Desktop akan menjadi pengamat kejadian selama penelitian berlangsung.
g. Kali Linux berperan sebagai penyerang dengan menggunakan berbagai metode penyerangan.

2.1.3. Design

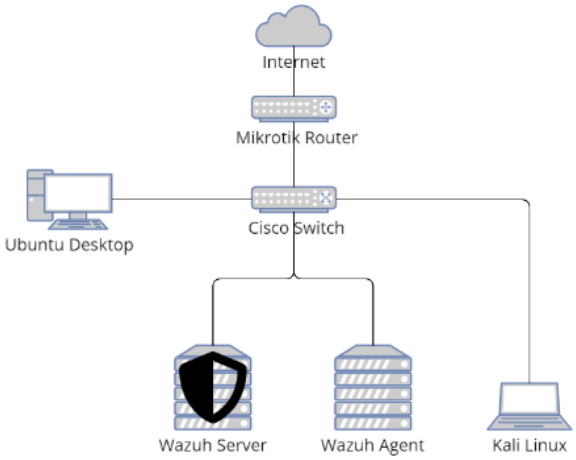
Setelah melewati tahap perencanaan, dibuatlah desain untuk penelitian yang akan dilakukan. Penelitian ini akan menguji kinerja IDPS Wazuh dalam 4 skenario yang berbeda.

a. Skenario 1
Pada skenario 1, Router akan terhubung dengan internet, Kali Linux dan Switch. Kemudian Switch akan dihubungkan dengan Ubuntu Desktop, Wazuh Server dan Wazuh Agent.



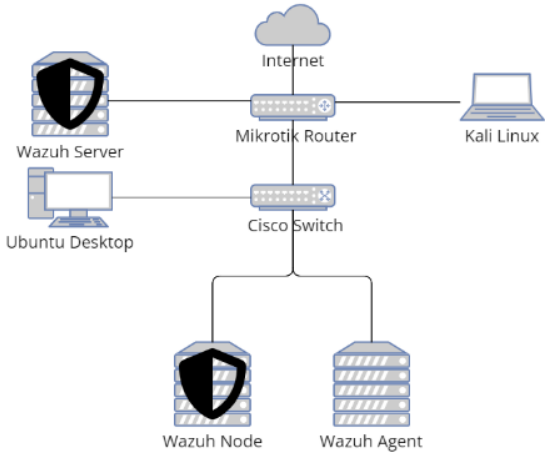
Gambar 1. Desain Skenario 1

b. Skenario 2
Pada skenario 2, Router akan terhubung dengan internet dan Switch. Kemudian Switch akan dihubungkan dengan Kali Linux, Ubuntu Desktop, Wazuh Server dan Wazuh Agent.



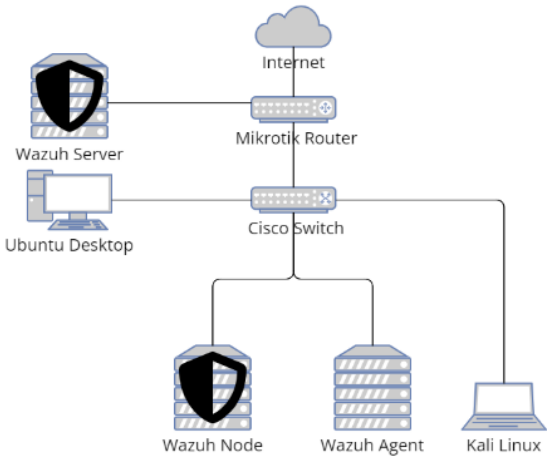
Gambar 2. Desain Skenario 2

c. Skenario 3
Pada skenario 3, Router dihubungkan dengan internet, Wazuh Server, Kali Linux dan Switch. Lalu Switch dihubungkan dengan Ubuntu Desktop, Wazuh Node, dan Wazuh Agent.



Gambar 3. Desain Skenario 3

d. Skenario 4
Pada skenario 4, Router dihubungkan dengan internet, Wazuh Server, dan Switch. Lalu Switch dihubungkan dengan Kali Linux, Ubuntu Desktop, Wazuh Node, dan Wazuh Agent.



Gambar 4. Desain Skenario 4

Kesimpulan dari keempat skenario di atas adalah:

a. Skenario 1 menempatkan Kali Linux sebagai penyerang pada *network* yang berbeda dengan Wazuh Server dan Wazuh Agent.
b. Skenario 2 menempatkan Kali Linux pada *network* yang sama dengan Wazuh Server dan Wazuh Agent
c. Skenario 3 menempatkan Kali Linux dan Wazuh Server pada *network* yang berbeda dengan Wazuh Agent
d. Skenario 4 menempatkan Kali Linux pada network yang berbeda dengan Wazuh Server dan pada *network* yang sama dengan Wazuh Agent.

2.1.4. Implement

Setelah mendesain sebuah topologi untuk penelitian, selanjutnya topologi tersebut akan dibuat pada lab virtual lalu dikonfigurasi sehingga dapat dijalankan.

2.1.5. Operate

Setelah dikonfigurasi, masing-masing skenario akan dijalankan dan diuji menggunakan 5 fitur keamanan dari Wazuh, yaitu:

a. File Integrity Monitoring
   Untuk menguji fitur ini, masing-masing Wazuh Agent akan dibuatkan sebuah file dummy yang akan diubah isinya lalu diamati dan diukur waktu reponsnya menggunakan standar TIPHON.
b. Bruteforce Detection
   Wazuh Agent pada masing-masing skenario akan diserang menggunakan metode *Bruteforce SSH* menggunakan hydra dari Kali Linux dan akan diamati kinerja deteksinya.
c. DoS Detection
   Wazuh Agent pada masing-masing skenario akan diserang menggunakan hping3 yang merupakan aplikasi yang dapat mengirimkan paket ICMP/UDP/TCP khusus dan dapat digunakan untuk menguji firewall [26], dan kemudian akan diamati kinerja deteksinya.
d. Port Scanning Detection
   Wazuh Agent masing-masing skenario akan menerima serangan Port Scanning menggunakan Nmap yang merupakan tool open-source untuk memindai celah dan port yang terbuka [27], lalu akan diamati kinerja deteksinya.
e. SQL Injection
   Wazuh Agent masing-masing skenario akan menerima serangan SQL Injection menggunakan curl dan akan diamati kinerja deteksinya.

2.1.6. Optimize

Pada tahap optimize, setiap fitur yang sudah berhasil dijalankan akan dioptimasi sehingga dapat mencapai tingkat efektivitas tertinggi yang dapat dicapai. Optimasi pada fitur File Integrity Monitoring yaitu dengan mempersempit direktori yang akan diamati sehingga lebih efisien lebih cepat dalam melaporkan perubahan. Kemudian optimasi untuk fitur lain seperti Bruteforce Detection, DoS Detection, Port Scanning Detection, dan SQL Injection yaitu dengan mengkonfigurasi sebuah *active response* dimana Wazuh dapat memblokir IP dari penyerang sehingga penyerang tidak lagi mendapatkan akses menuju target.

2.2. Quality of Services (QoS)

Penelitian ini menggunakan parameter *Quality of Services* (*QoS*) sebagai standar untuk mengkategorikan kualitas layanan. *QoS* menurut Wulandari [28] adalah kemampuan sebuah layanan dalam memastikan, menjamin performa, serta merupakan indikator untuk menilai kualitas sebuah layanan. *European Telecommunication Standart Institute (ETSI)* mengeluarkan standar untuk QoS dengan nama TIPHON [29]. Menurut TIPHON, ada 4 parameter QoS, antara lain:

2.2.1. Throughput

Throughput adalah kecepatan transfer data efektif dan diukur menggunakan satuan *bit per second (bps).* Pengkategorian Throughput dapat dilihat pada tabel di bawah ini.

Tabel 1. Pengukuran Throughput

Indeks	Throughput (bps)	Kategori
4	100	Sangat Baik
3	75	Baik
2	50	Sedang
1	<25	Buruk

Berdasarkan tabel di atas, Throughput dikategorikan Sangat Baik jika mencapai 100 bps, Baik jika mencapai 75 bps, Sedang jika mencapai 50 bps, dan buruk jika kurang dari 25 bps.

2.2.2. Packet Loss

Packet Loss adalah parameter yang menunjukkan banyaknya jumlah *packet* yang hilang ketika proses transfer data terjadi. Pengkategorian Packet Loss dapat dilihat pada tabel di bawah ini.

Tabel 2. Pengukuran Packet Loss

Indeks	Packet Loss (%)	Kategori
4	0	Sangat Baik
3	3	Baik
2	15	Sedang
1	25	Buruk

Berdasarkan tabel di atas, Packet Loss dikategorikan Sangat Baik jika mencapai 0%, Baik jika mencapai 3%, Sedang jika mencapai 15%, dan buruk jika mencapai 25%.

2.2.3. Delay

Delay adalah waktu yang dibutuhkan oleh sebuah data untuk bergerak ke tujuannya. Pengkategorian Delay dapat dilihat pada tabel di bawah ini.

Tabel 3. Pengukuran Delay

Indeks	Delay (ms)	Kategori
4	<150	Sangat Baik
3	150 – 300	Baik
2	300 – 450	Sedang
1	>450	Buruk

Berdasarkan tabel di atas, Delay dikatakan Sangat Baik jika kurang dari 150 ms, Baik jika mencapai 150 – 300 ms, Sedang jika mencapai 300 – 450 ms, dan buruk jika lebih dari 450 ms.

2.2.4. Jitter

Jitter adalah variasi kedatangan paket yang diakibatkan oleh antrean dan waktu pengolahan data yang berbeda. Pengkategorian Jitter dapat dilihat pada tabel di bawah ini.

Tabel 4. Pengukuran Jitter

Indeks	Jitter (ms)	Kategori
4	0	Sangat Baik
3	<75	Baik
2	75 – 125	Sedang
1	125 – 225	Buruk

Berdasarkan tabel di atas, Jitter dikatakan Sangat Baik jika mencapai 0 ms, Baik jika mencapai 75 ms, Sedang jika mencapai 75 – 125 ms, dan buruk jika mencapai 125 – 225 ms.

3. HASIL DAN PEMBAHASAN

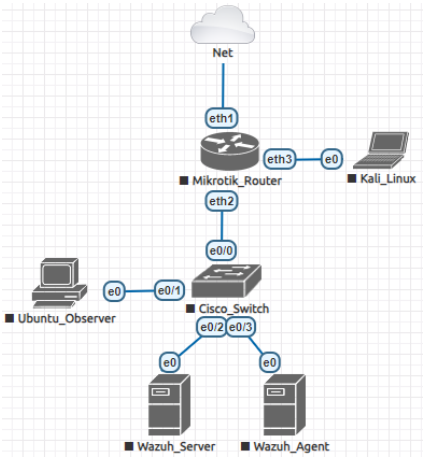
Setelah desain topologi dibuat, maka selanjutnya pada tahap *Implement*, topologi akan diwujudkan dan dikonfigurasi. Selanjutnya pada tahap *Operate*, topologi yang sudah diimplementasi akan dijalankan dan diuji, lalu pada tahap *Optimize* akan dilakukan optimasi sehingga mendapatkan hasil yang efektif.

3.1. Implement

Pada tahap ini, masing-masing skenario akan dikonfigurasi di dalam lab virtual EVE-NG sehingga dapat dijalankan.

a. Skenario 1

Pada skenario 1, Kali Linux akan ditempatkan pada *network* yang berbeda dengan Wazuh Server, Ubuntu Desktop, dan Wazuh Agent.



Gambar 5. Topologi Skenario 1

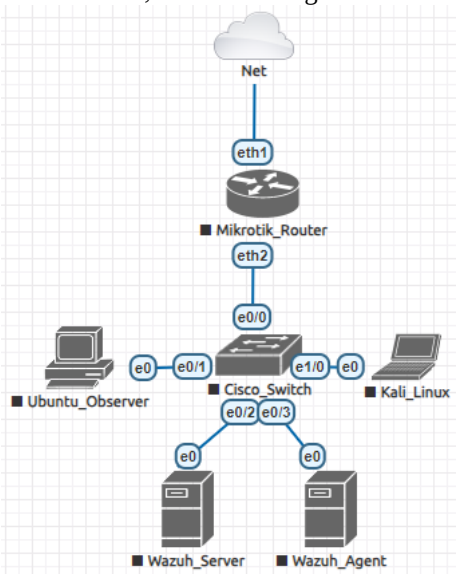
Spesifikasi dan pengalamatan IP bisa dilihat pada tabel di bawah ini.

Tabel 5. Spesifikasi Skenario 1

Nama	CPU	RAM	Alamat IP
Mikrotik_Router	1	256 MB	192.168.195.129 192.168.69.1 192.168.76.1
Cisco_Switch	-	1024 KB NVRAM 512 MB RAM	-
Kali_Linux	2	1024 MB	192.168.76.100
Ubuntu_Observer	2	1024 MB	192.168.69.100
Wazuh_Server	2	4096 MB	192.168.69.69
Wazuh_Agent	2	1024 MB	192.168.69.254

b. Skenario 2

Pada skenario 2, Kali Linux ditempatkan pada *network* yang sama dengan Ubuntu Desktop, Wazuh Server, dan Wazuh Agent.



Gambar 6. Topologi Skenario 2

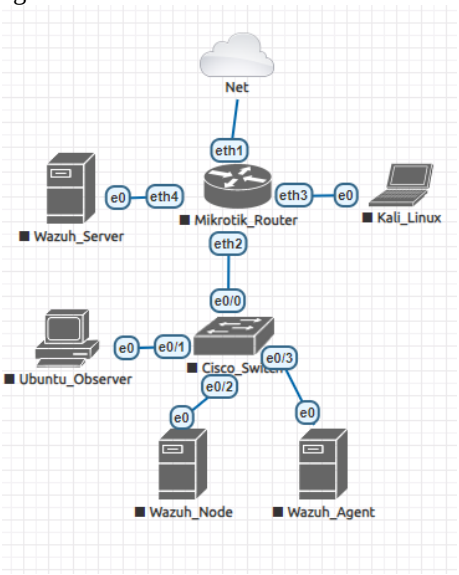
Spesifikasi dan pengalamatan IP dapat dilihat pada tabel di bawah ini.

Tabel 6. Spesifikasi Skenario 2

Nama	CPU	RAM	Alamat IP
Mikrotik_Router	1	256 MB	192.168.195.129 192.168.69.1
Cisco_Switch	-	1024 KB NVRAM 512 MB RAM	-
Kali_Linux	2	1024 MB	192.168.76.100
Ubuntu_Observer	2	1024 MB	192.168.69.100
Wazuh_Server	2	4096 MB	192.168.69.69
Wazuh_Agent	2	1024 MB	192.168.69.254

c. Skenario 3

Pada skenario 3, Kali Linux dan Wazuh Server ditempatkan pada *network* yang berbeda dengan Ubuntu Desktop, Wazuh Node, dan Wazuh Agent.



Gambar 7. Topologi Skenario 3

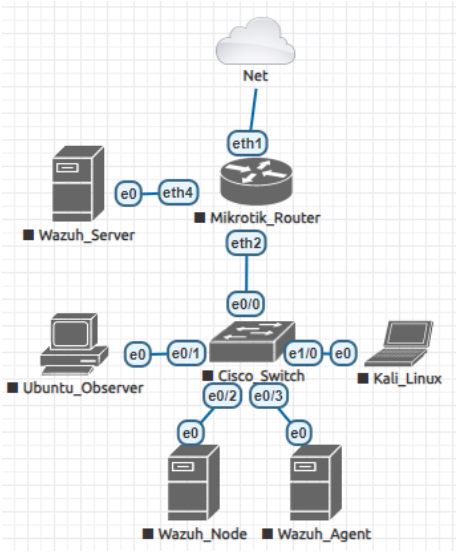
Spesifikasi dan pengalamatan IP dapat dilihat pada tabel di bawah ini.

Tabel 7. Spesifikasi Skenario 3

Nama	CPU	RAM	Alamat IP
Mikrotik_Router	1	256 MB	192.168.195.129 192.168.69.1 192.168.76.1 192.168.86.1
Cisco_Switch	-	1024 KB NVRAM 512 MB RAM	-
Kali_Linux	2	1024 MB	192.168.76.100
Ubuntu_Observer	2	1024 MB	192.168.69.100
Wazuh_Node	2	4096 MB	192.168.69.69
Wazuh_Server	2	4096 MB	192.168.86.86
Wazuh_Agent	2	1024 MB	192.168.69.254

d. Skenario 4

Pada skenario 4, Kali Linux ditempatkan pada *network* yang berbeda dengan Wazuh Server, namun pada *network* yang sama dengan Ubuntu Desktop, Wazuh Node, dan Wazuh Agent.



Gambar 8. Topologi Skenario 4

Spesifikasi dan pengalamatan IP dapat dilihat pada tabel di bawah ini.

Tabel 8. Spesifikasi Skenario 4

Nama	CPU	RAM	Alamat IP
Mikrotik_Router	1	256 MB	192.168.195.129 192.168.69.1 192.168.86.1
Cisco_Switch	-	1024 KB NVRAM 512 MB RAM	-
Kali_Linux	2	1024 MB	192.168.69.200
Ubuntu_Observer	2	1024 MB	192.168.69.100
Wazuh_Node	2	4096 MB	192.168.69.69
Wazuh_Server	2	4096 MB	192.168.86.86
Wazuh_Agent	2	1024 MB	192.168.69.254

Setelah konfigurasi selesai dilakukan, maka tahap *Operate* dapat dilakukan.

3.2. Operate

Pada tahap ini akan dilakukan pengujian fitur pada masing-masing skenario. Fitur yang diuji antara lain, seperti File Integrity Monitoring, Bruteforce Detection, DoS Detection, Port Scanning Detection, dan SQL Injection Detection. Hasil dari pengujian berupa selisih waktu deteksi antara sistem target dan Wazuh.

3.2.1. File Integrity Monitoring

Dalam pengujian fitur ini, Wazuh Agent pada masing-masing skenario akan dibuatkan sebuah file dummy lalu akan diubah isinya. Setelah itu selisih waktu deteksi antara sistem Wazuh Agent dan Wazuh Server akan diamati. Daftar perintah yang digunakan beserta fungsinya dapat dilihat pada tabel di bawah ini.

Tabel 9. Perintah dalam Pengujian Fitur File Integrity Monitoring

Perintah	Fungsi
touch coba	Membuat file dummy bernama coba pada Wazuh Agent
nano coba	Membuka file coba menggunakan aplikasi nano pada Wazuh Agent dan mengubah isinya
ls –full-time	Melihat waktu modifikasi yang tercatat pada sistem Wazuh Agent

Setelah menjalankan perintah di atas, maka akan diperoleh waktu yang tercatat pada sistem Wazuh Agent. Kemudian waktu tersebut dibandingkan denga waktu yang tercatat pada Wazuh Server yang dapat dilihat melalui Dashboard.

Gambar 9. Selisih Waktu antara Wazuh Agent dan Wazuh Server pada Fitur File Integrity Monitoring

Pengujian ini dilakukan pada masing-masing skenario sehingga didapatkan hasil pengukuran seperti berikut.

Tabel 10. Hasil Pengukuan Kinerja File Integrity Monitoring

Skenario	Delay (ms)	
	Pembuatan File	Modifikasi File
1	45	40
2	38	16
3	10	4
4	25	17

3.2.2. Bruteforce Detection

Dalam pengujian fitur ini, Wazuh Agent akan diserang oleh Kali Linux menggunakan *hydra*. Sebelumnya akan dibuat sebuah file *txt* yang berisi daftar password yang akan dicoba selama penyerangan. Daftar perintah beserta fungsinya dapat dilihat pada tabel di bawah ini.

Tabel 11. Perintah dalam Pengujian Bruteforce Detection

Perintah	Fungsi
nano pass.txt	Membuat file pass.txt pada Kali Linux yang kemudian diisi daftar password yang akan digunakan dalam penyerangan
hydra -l <username> -P pass.txt <IP Target> ssh	Menjalankan hydra pada Kali Linux menggunakan username dan daftar password yang dipilih untuk memulai serangan terhadap IP target melalui protokol SSH

Setelah perintah di atas dijalankan pada Kali Linux, maka serangan akan terdeteksi oleh Wazuh Server dan terlihat pada Dashboard.

Gambar 10. Serangan Bruteforce Tercatat Pada Wazuh Server

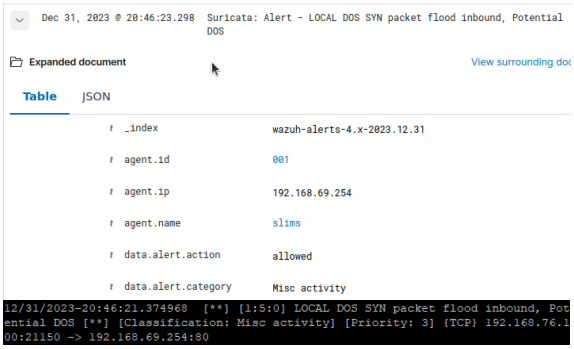
3.2.3. DoS Detection

Dalam pengujian fitur ini, Wazuh Agent akan diserang oleh Kali Linux menggunakan hping3. penyerangan akan dijalankan selama 10 detik lalu serangan akan dihentikan. Kemudian selisih waktu antara Wazuh Agent dan Wazuh Server akan diamati. Daftar perintah yang digunakan beserta fungsinya dapat dilihat pada tabel dibawah.

Tabel 12. Perintah dalam Pengujian DoS Detection

Perintah	Fungsi
hping3 -S –flood -V -p 80 <IP Target>	Menjalankan hping3 pada Kali Linux untuk melakukan penyerangan *flooding* terhadap port 80 milik target
tail -f /var/log/suricata/fast.log	Memantau aktivitas *log* pendeteksi serangan DoS pada Wazuh Agent

Setelah penyerangan berlangsung selama 10 detik, penyerangan dihentikan dan selisih waktu antara *log* pada Wazuh Agent dan Wazuh Server akan dibandingkan.



Gambar 11. Selisih Waktu antara Wazuh Agent dan Wazuh Server pada fitur DoS Detection

Pengujian dilakukan pada masing-masing skenario sehingga mendapatkan hasil seperti pada tabel berikut.

Tabel 13. Hasil Pengukuran Kinerja DoS Detection

Skenario	Delay (ms)
1	1.924
2	1.117
3	2.353
4	584

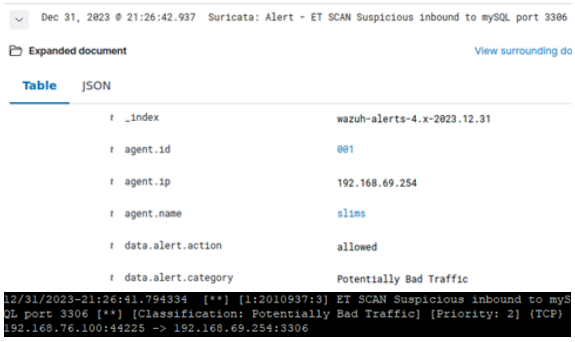
3.2.4. Port Scanning Detection

Dalam pengujian fitur ini, Wazuh Agent akan diserang oleh Kali Linux menggunakan Nmap. Penyerangan ini akan mendeteksi celah dan port terbuka dari target. Kemudian selisih waktu antara Wazuh Agent dan Wazuh Server akan diamati. Daftar perintah yang digunakan beserta fungsinya dapat dilihat pada tabel dibawah.

Tabel 14. Perintah dalam Pengujian Port Scanning Detection

Perintah	Fungsi
nmap -sS <IP Target>	Menjalankan Nmap pada Kali Linux untuk memindai dan mencari celah pada port terbuka milik target
tail -f /var/log/suricata/fast.log	Memantau aktivitas *log* pendeteksi serangan Port Scanning pada Wazuh Agent

Setelah Nmap selesai memindai, selisih waktu antara *log* pada Wazuh Agent dan Wazuh Server akan dibandingkan.



Gambar 11. Selisih Waktu antara Wazuh Agent dan Wazuh Server pada fitur Port Scanning Detection

Pengujian dilakukan pada masing-masing skenario sehingga mendapatkan hasil seperti pada tabel berikut.

Tabel 15. Hasil Pengukuran Kinerja Port Scanning Detection

Skenario	Delay (ms)
1	1.143
2	230
3	2027
4	801

3.2.5. SQL Injection Detection

Dalam pengujian fitur ini, Wazuh Agent akan diserang oleh Kali Linux menggunakan curl. Kemudian selisih waktu antara Wazuh Agent dan Wazuh Server akan diamati. Daftar perintah yang digunakan beserta fungsinya dapat dilihat pada tabel dibawah.

Tabel 16. Perintah dalam Pengujian SQL Injection Detection

Perintah	Fungsi
curl -XGET "http://<IP Target>/users/?id=SELECT+*+FROM+users";	Menjalankan curl pada Kali Linux untuk menginjeksi *Query* ke dalam situs target
tail -f /var/log/apache2/access.log	Memantau aktivitas *log* pendeteksi serangan SQL Injection pada Wazuh Agent

Setelah serangan selesai dilakukan, selisih waktu antara *log* pada Wazuh Agent dan Wazuh Server akan dibandingkan.

Gambar 11. Selisih Waktu antara Wazuh Agent dan Wazuh Server pada fitur SQL Injection Detection

Pengujian dilakukan pada masing-masing skenario sehingga mendapatkan hasil seperti pada tabel berikut.

Tabel 17. Hasil Pengukuran Kinerja SQL Injection Detection

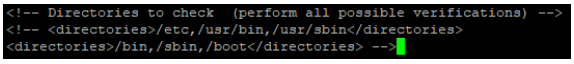
Skenario	Delay (ms)
1	1.387
2	1.880
3	326
4	455

3.3. Optimize

Pada tahap ini dilakukan optimasi terhadap fitur-fitur yang telah diujikan. Optimasi ini bertujuan untuk membuat fitur lebih efektif dan efisien dalam mendeteksi serta mencegah serangan jaringan.

3.3.1. File Integrity Monitoring

Dalam mengoptimasi fitur File Integrity Monitoring, konfigurasi Wazuh Agent yang terletak pada "*/var/ossec/etc/ossec.conf*" akan diubah dengan menambahkan baris "*<directories check_all="yes" report_changes="yes" realtime="yes" whodata="yes">/home/slims</directories>*" setelah baris yang ada pada gambar di bawah ini.



Gambar 12. Konfigurasi File Integrity Monitoring

Dengan ditambahkannya baris di atas, maka fitur File Integrity Monitoring hanya akan berfokus dalam memantau direktori "*/home/slims*". Hal ini bisa meningkatkan efektivitas karena sistem tidak perlu memantau ke setiap direktori yang ada pada perangkat, melainkan hanya ke dalam direktori yang sudah ditentukan.

3.3.2. Bruteforce Detection

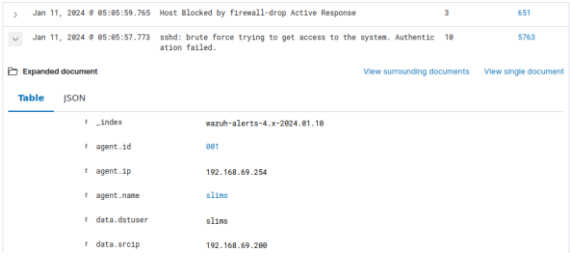
Dalam mengoptimasi fitur Bruteforce detection, Wazuh Server akan dikonfigurasi sehingga jika ada penyerang yang menggunakan

metode Bruteforce akan menyebabkan IP dari penyerang tersebut akan diblokir oleh Wazuh Agent, baik sementara waktu maupun permanen. Optimasi dilakukan pada file "*/var/ossec/etc/ossec.conf*" pada Wazuh Server dengan cara menambahkan baris seperti pada gambar di bawah ini.

```
<active-response>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>5763</rules_id>
  <timeout>180</timeout>
</active-response>
```

Gambar 13. Konfigurasi Optimasi Fitur Bruteforce Detection

Konfigurasi di atas menunjukkan bahwa jika ada serangan yang melanggar *rules* dengan ID 5763 yaitu deteksi Bruteforce, maka IP dari penyerang tersebut akan diblokir selama 180 detik atau 3 menit. Hasil dari optimasi tersebut dapat dilihat pada gambar di bawah.



Gambar 14. Hasil Optimasi Fitur Bruteforce Detection

3.3.3. DoS Detection

Dalam mengoptimasi fitur DoS detection, Wazuh Server akan dikonfigurasi sehingga jika ada serangan DoS akan menyebabkan IP dari penyerang tersebut akan diblokir oleh Wazuh Agent, baik sementara waktu maupun permanen. Optimasi dilakukan pada file "*/var/ossec/etc/ossec.conf*" pada Wazuh Server dengan cara menambahkan baris seperti pada gambar di bawah ini.

```
<active-response>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>100200</rules_id>
  <timeout>180</timeout>
</active-response>
```

Gambar 15. Konfigurasi Optimasi Fitur DoS Detection

Konfigurasi di atas menunjukkan bahwa jika ada serangan yang melanggar *rules* dengan ID 100200 yaitu deteksi DoS, maka IP dari penyerang tersebut akan diblokir selama 180 detik atau 3 menit. Hasil dari optimasi tersebut dapat dilihat pada gambar di bawah.

Gambar 16. Hasil Optimasi Fitur DoS Detection

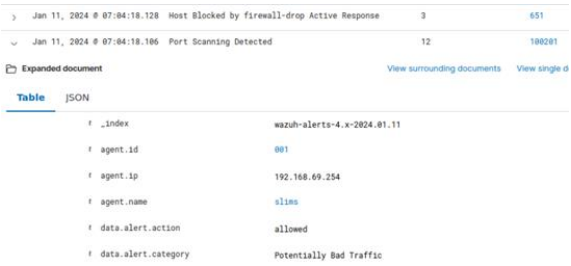
3.3.4. Port Scanning Detection

Dalam mengoptimasi fitur Port Scanning detection, Wazuh Server akan dikonfigurasi sehingga jika ada penyerang yang memindai port dan celah yang terbuka akan menyebabkan IP dari penyerang tersebut akan diblokir oleh Wazuh Agent, baik sementara waktu maupun permanen. Optimasi dilakukan pada file "*/var/ossec/etc/ossec.conf*" pada Wazuh Server dengan cara menambahkan baris seperti pada gambar di bawah ini.

```
<active-response>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>100201</rules_id>
  <timeout>180</timeout>
</active-response>
```

Gambar 17. Konfigurasi Optimasi Fitur Port Scanning Detection

Konfigurasi di atas menunjukkan bahwa jika ada serangan yang melanggar *rules* dengan ID 100201 yaitu deteksi Port Scanning, maka IP dari penyerang tersebut akan diblokir selama 180 detik atau 3 menit. Hasil dari optimasi tersebut dapat dilihat pada gambar di bawah.



Gambar 18. Hasil Optimasi Fitur Bruteforce Detection

3.3.5. SQL Injection Detection

Dalam mengoptimasi fitur SQL Injection detection, Wazuh Server akan dikonfigurasi sehingga jika ada penyerang yang mencoba melakukan SQL Injection akan menyebabkan IP dari penyerang tersebut akan diblokir oleh Wazuh Agent, baik sementara waktu maupun permanen. Optimasi dilakukan pada file "*/var/ossec/etc/ossec.conf*" pada Wazuh Server dengan cara menambahkan baris seperti pada gambar di bawah ini.

```
<active-response>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>100202</rules_id>
  <timeout>180</timeout>
</active-response>
```

Gambar 19. Konfigurasi Optimasi Fitur SQL Injection Detection

Konfigurasi di atas menunjukkan bahwa jika ada serangan yang melanggar *rules* dengan ID 100202 yaitu deteksi SQL Injection, maka IP dari penyerang tersebut akan diblokir selama 180 detik atau 3 menit. Hasil dari optimasi tersebut dapat dilihat pada gambar di bawah.

Gambar 20. Hasil Optimasi Fitur SQL Injection Detection

4. DISKUSI

Mengimplementasikan Wazuh sebagai IDPS memberikan kemudahan bagi pengguna untuk memantau jaringan, analisis *log*, kemudahan konfigurasi, dan fitur keamanan yang efektif.

4.1. Pengukuran QoS

Hasil pengukuran delay setiap fitur pada masing-masing skenario kemudian akan dikategorikan menggunakan standar QoS dari ETSI.

4.1.1. File Integrity Monitoring

Setelah file dummy dibuat lalu diubah, selisih waktu antara sistem Wazuh Agent dan Wazuh Server dibandingkan dan diukur menggukanan standar TIPHON.

Tabel 18. Pengukuran Kinerja File Integrity Monitoring

Skenario	Delay (ms)		Kategori
	Pembuatan File	Modifikasi File	
1	45	40	Sangat Baik
2	38	16	Sangat Baik
3	10	4	Sangat Baik
4	25	17	Sangat Baik

Berdasarkan standar TIPHON dari ETSI, Delay pada semua skenario termasuk dalam kategori sangat baik karena kurang dari 150 ms.

4.1.2. DoS Detection

Setelah serangan DoS selama 10 detik dihentikan, selisih waktu antara sistem Wazuh Agent dan Wazuh Server dibandingkan dan diukur menggukanan standar TIPHON.

Tabel 19. Pengukuran Kinerja DoS Detection

Skenario	Delay (ms)	Kategori
1	1.924	Buruk
2	1.117	Buruk
3	2.353	Buruk
4	584	Buruk

Berdasarkan standar TIPHON dari ETSI, Delay pada semua skenario termasuk dalam kategori buruk karena lebih dari 450 ms.

4.1.3. Port Scanning Detection

Setelah serangan Port Scanning selesai, selisih waktu antara sistem Wazuh Agent dan Wazuh Server dibandingkan dan diukur menggukanan standar TIPHON.

Tabel 20. Pengukuran Kinerja Port Scanning Detection

Skenario	Delay (ms)	Kategori
1	1.143	Buruk
2	230	Baik
3	2.027	Buruk
4	801	Buruk

Berdasarkan standar TIPHON dari ETSI, Delay pada skenario 2 termasuk dalam kategori baik karena kurang dari 300 ms. Sedangkan pada skenario lain termasuk buruk karena bernilai lebih dari 450ms.

4.1.4. SQL Injection Detection

Setelah serangan SQL Injection selesai, selisih waktu antara sistem Wazuh Agent dan Wazuh Server dibandingkan dan diukur menggukanan standar TIPHON.

Tabel 21. Pengukuran Kinerja SQL Injection Detection

Skenario	Delay (ms)	Kategori
1	1.387	Buruk
2	1.880	Buruk
3	326	Sedang
4	801	Buruk

Berdasarkan standar TIPHON dari ETSI, Delay pada skenario 3 termasuk dalam kategori sedang karena kurang dari 450 ms. Sedangkan pada skenario lain termasuk buruk karena bernilai lebih dari 450ms.

4.2. Faktor yang Mempengaruhi Nilai QoS

Beberapa faktor mungkin dapat mempengaruhi nilai QoS dalam penelitian ini, diantaranya:

4.2.1. Alokasi Resource

Penelitian ini menggunakan virtual lab yang berjalan di dalam host Windows 10 dan menjalankan banyak *nodes* di dalamnya. Sehingga memiliki kemungkinan jika *resource* yang dimiliki host akan terbagi ke masing-masing *node* secara tidak merata. Hal ini dapat mempengaruhi *QoS* dikarenakan *resource* yang seharusnya sudah ditetapkan untuk mengirimkan data bisa saja digunakan oleh proses lain, sehingga terjadi kenaikan delay.

4.2.2. Lingkungan Percobaan

Penelitian ini berjalan menggunakan *Hosted Virtualization* yang merupakan virtualisasi tipe 2. Hal ini berarti virtual lab ini berjalan pada host yang sudah memiliki OS. Hal ini menyebabkan terbaginya

resource dan memicu kenaikan delay. Graniszewski & Arciszewski [30] melakukan uji performa terharap *Bare-Metal Virtualization* (Tipe 1) dan *Hosted Virtualization* (Tipe 2). Pengujian tersebut meliputi uji *throughput*, *kernel compilation*, dan *CPU benchmark*. Hasil dari pengujian tersebut menunjukkan bahwa virtualisasi tipe 1 memiliki hasil yang lebih unggul daripada tipe 2. Hal ini disebabkan karena virtualisasi tipe 1 memiliki akses penuh menuju *resource* yang ada pada *hardware*, sedangan virtualisasi tipe 2 masih harus membagi *resource* dengan host ditempatnya dijalankan.

5. KESIMPULAN

Dari pengujian keempat skenario yang telah dibuat, Wazuh sebagai IDPS terbukti dapat dengan efektif mendeteksi dan mencegah berbagai macam serangan jaringan, baik dari dalam maupun luar *network*. Nilai delay yang tinggi disebabkan oleh virtualisasi yang menggunakan tipe 2, sehingga *resource* yang digunakan tidak dapat terbagi secara optimal.

Pemodelan IDPS menggunakan platform Wazuh menghasilkan kesimpulan bahwa Wazuh sebagai IDPS *open-source* dapat diandalkan oleh organisasi yang belum memiliki rencana untuk mendeteksi dan mencegah berbagai macam serangan jaringan karena memiliki fitur seperti *active response* yang dapat dikonfigurasi sesuai dengan kebutuhan pengguna, penampilan log yang mudah dibaca oleh pengguna dan tentunya gratis.

DAFTAR PUSTAKA

[1] Z. Tie, "A mobile agent-based system for server resource monitoring," *Cybernetics and Information Technologies*, vol. 13, no. 4, pp. 104–117, 2013, doi: 10.2478/cait-2013-0057.

[2] A. Raharjo, R. W. Bintoro, N. A. Tri Utami, and M. Suzuki, "The Legal Policy of Criminal Justice Bureaucracy Cybercrime," *BESTUUR*, vol. 10, no. 2, p. 105, Dec. 2022, doi: 10.20961/bestuur.v10i2.64498.

[3] C. Sirois, "New McAfee Report Estimates Global Cybercrime Losses to Exceed $1 Trillion." [Online]. Available: https://bit.ly/3imW1jv

[4] Iskandar, "Situs Pemantauan Virus Corona DKI Jakarta Sempat Kena Serangan DDoS, Warganet Murka." [Online]. Available: https://bit.ly/3GV5qbh

[5] W. K. Pertiwi, "BPJS Kesehatan Akui Ada Kemungkinan Peretasan Data 279 Juta Warga RI." [Online]. Available: https://bit.ly/3VZVdPg

[6] P. Stavroulakis and M. Stamp, *Handbook of Information and Communication Security*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. doi: 10.1007/978-3-642-04117-4.

[7] A. L. Imoize, T. Oyedare, M. E. Otuokere, and S. Shetty, "Software Intrusion Detection Evaluation System: A Cost-Based Evaluation of Intrusion Detection Capability," *Communications and Network*, vol. 10, no. 04, pp. 211–229, 2018, doi: 10.4236/cn.2018.104017.

[8] H. Sjölinder and O. Carlsson, "Open Source Software Licenses Impact on Businesses," 2023. [Online]. Available: www.bth.se

[9] S. N. Kumar, "Review on Network Security and Cryptography," *International Transaction of Electrical and Computer Engineers System*, vol. 3, no. 1, pp. 1–11, 2015, doi: 10.12691/iteces-3-1-1.

[10] M. Syafrizal, "ISO 17799: Standar Sistem Manajemen Keamanan Informasi," 2007.

[11] R. Ali, A. Ali, F. Iqbal, M. Hussain, and F. Ullah, "Deep Learning Methods for Malware and Intrusion Detection: A Systematic Literature Review," *Security and Communication Networks*, vol. 2022. Hindawi Limited, 2022. doi: 10.1155/2022/2959222.

[12] P. Maniriho, A. N. Mahmood, and M. J. M. Chowdhury, "A systematic literature review on Windows malware detection: Techniques, research issues, and future directions," *Journal of Systems and Software*, vol. 209, p. 111921, Mar. 2024, doi: 10.1016/j.jss.2023.111921.

[13] N. N. Abdulla and R. K. Hasoun, "Review of Detection Denial of Service Attacks using Machine Learning through Ensemble Learning," *Iraqi Journal for Computers and Informatics*, vol. 48, no. 1, pp. 13–20, 2022.

[14] A. Cheema, M. Tariq, A. Hafiz, M. M. Khan, F. Ahmad, and M. Anwar, "Prevention Techniques against Distributed Denial of Service Attacks in Heterogeneous Networks: A Systematic Review," *Security and Communication Networks*, vol. 2022. Hindawi Limited, 2022. doi: 10.1155/2022/8379532.

[15] A. Alazzawi, "SQL INJECTION DETECTION USING RNN DEEP LEARNING MODEL," *Journal of Applied Engineering and Technological Science*, vol. 5, no. 1, pp. 531–541, 2023.

[16] G. Kostopoulos, *Cyberspace and Cybersecurity*, vol. Second edition. Boca Raton, Florida: Auerbach Publications, 2017. [Online]. Available: http://e-resources.perpusnas.go.id:2048/login?url

=https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1620732&site=eds-live

[17] Wazuh Documentation, "Getting Started." Accessed: Jan. 09, 2024. [Online]. Available: https://documentation.wazuh.com/4.2/getting-started/index.html

[18] Wazuh Documentation, "Wazuh Components." Accessed: Jan. 09, 2024. [Online]. Available: https://documentation.wazuh.com/4.3/getting-started/components/index.html

[19] T. Ernawati and F. Rachmat, "Keamanan Jaringan dengan Cowrie Honeypot dan Snort Inline-Mode sebagai Intrusion Prevention System," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 180–186, Feb. 2021, doi: 10.29207/resti.v5i1.2825.

[20] Harahap A and Hutrianto, "INTRUSION DETECTION AND ANOMALY MENGGUNAKAN WAZUH PADA UNIVERSITAS MUHAMMADIYAH PALEMBANG," *Bina Darma Conference on Computer Science*, pp. 324–328, Nov. 2021.

[21] M. S. S. Husain, L. Fid Aksara, and N. Ransi, "IMPLEMENTASI KEAMANAN SERVER PADA JARINGAN WIRELESS MENGGUNAKAN METODE INTRUSION DETECTION AND PREVENTION SYSTEM (IDPS) (STUDI KASUS : TECHNO'S STUDIO)," *semanTIK*, vol. 4, no. 2, pp. 11–20, 2018, doi: 10.5281/zenodo.1407864.

[22] N. S. S. Yomo, A. Z. Mardiansyah, and I. W. A. Arimbawa, "Deteksi Serangan SQL Injection Menggunakan Security Information and Event Management (SIEM) Wazuh (Sudi Kasus: Sistem Informasi Akademik Universitas Mataram)," 2023.

[23] M. D. Pratama, F. Nova, and D. Prayama, "Wazuh sebagai Log Event Management dan Deteksi Celah Keamanan pada Server dari Serangan Dos," *JITSI : Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 3, no. 1, pp. 1–7, Mar. 2022, doi: 10.30630/jitsi.3.1.59.

[24] Verawati, "Merancang dan Membangun Jaringan VLAN Dengan Metode RIP pada Dinas Sosial dan Tenaga Kerja Menggunakan Cisco Router," *Jurnal Cendikia*, vol. 12, no. 1, pp. 23–30, Apr. 2016.

[25] StatCounter, "Desktop Operating System Market Share Worldwide 2021 - 2023." Accessed: Jan. 25, 2024. [Online]. Available: https://gs.statcounter.com/os-market-share/desktop/worldwide/#yearly-2021-2023-bar

[26] F. I. Tampati, F. G. Setyawan, W. W. Sejati, and A. R. Kardian, "Analisis Perbandingan Performa CPU pada Sistem Operasi FreeBSD 64-bit dan RedHat Linux 64-bit terhadap Serangan Denial of Service (DoS) Menggunakan Hping3," *CESS (Journal of Computing Engineering, System and Science)*, vol. 8, no. 1, pp. 209–219, 2023, [Online]. Available: www.jurnal.unimed.ac.id

[27] Y. Singh, P. Singh, G. Sinha, and G. Sinha, "Footprinting Using Nmap," *Journal of Informatics Electrical and Electronics Engineer-ing*, vol. 03, pp. 1–15, 2022, doi: 10.54060/JIEEE/003.

[28] P. Wulandari, S. Soim, and M. Rose, "MONITORING DAN ANALISIS QOS (QUALITY OF SERVICE) JARINGAN INTERNET PADA GEDUNG KPA POLITEKNIK NEGERI SRIWIJAYA DENGAN METODE DRIVE TEST," *Prosiding SNATIF Ke-4 Tahun 2017*, 2017.

[29] ETSI, "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); General aspects of Quality of Service (QoS)," 1999. [Online]. Available: http://www.etsi.org

[30] W. Graniszewski and A. Arciszewski, "Performance analysis of selected hypervisors (Virtual Machine Monitors-VMMs)," *International Journal of Electronics and Telecommunications*, vol. 62, no. 3, pp. 231–236, Sep. 2016, doi: 10.1515/eletel-2016-0031.