Analisis Anomali Lalu Lintas Jaringan menggunakan Metode Prophet melalui Integrasi SNORT dan Telegram

Reinaldy Putra Simanulang[1*], Endyk Noviyantono[2], Okky Herodion Simung[3]

[123*]Teknik Informatika, STMIK PPKIA Tarakanita Rahmawati, Tarakan, Kalimantan Utara
Email: [1*]2151007@student.ppkia.ac.id, [2]endyk@ppkia.ac.id, [3]okky@ppkia.ac.id

Abstrak

Perkembangan teknologi komunikasi membawa tantangan signifikan terhadap keamanan jaringan komputer. Jaringan komputer rentan terhadap berbagai serangan siber seperti malware, DDoS, phishing, dan eksploitasi kerentanan keamanan. Sistem Deteksi Intrusi (IDS) adalah solusi keamanan jaringan yang dapat digunakan untuk mendeteksi dan menganalisis aktivitas mencurigakan dalam lalu lintas jaringan. Snort adalah IDS dengan kode sumber terbuka yang mampu mendeteksi berbagai ancaman menggunakan aturan berbasis tanda tangan dan analisis perilaku. Snort telah memiliki mekanisme khusus untuk melakukan pemblokiran alamat IP penyerang. Pada penelitian ini mengusulkan metode Prophet dari Facebook yang dapat digunakan sebagai mekanisme tambahan untuk melakukan blokir alamat penyerang IP saat ditemukan anomali jumlah serangan terhadap layanan Web, FTP, dan SSH. Hasil akhir penelitian menunjukkan bahwa pengelola jaringan mendapatkan notifikasi serangan melalui aplikasi Telegram, serta server memiliki peningkatan keamanan melalui mekanisme tambahan pada Snort untuk memblokir alamat penyerang IP.

Kata Kunci: analisis anomali, keamanan jaringan, ids, snort, telegram.

Network Traffic Anomaly Analysis using Prophet Method through SNORT and Telegram Integration

Abstract

The development of communication technology brings significant challenges to computer network security. Computer networks are vulnerable to various cyberattacks such as malware, DDoS, phishing, and exploiting security vulnerabilities. An Intrusion Detection System (IDS) is a network security solution that can be used to detect and analyze suspicious activity in network traffic. Snort is an open-source IDS capable of detecting various threats using signature-based rules and behavioral analysis. Snort has a special mechanism for blocking attacker IP addresses. This study proposes the Prophet method from Facebook that can be used as an additional mechanism to block attacker IP addresses when an anomaly in the number of attacks on Web, FTP, and SSH services is detected. The final results of the study show that network managers receive attack notifications via the Telegram application, and servers have increased security through an additional mechanism in Snort to block attacker IP addresses.

Keywords: *snort, telegram, notification, network security, intrusion detection system (IDS).*

I. PENDAHULUAN

Perkembangan jaringan komputer telah mengalami transformasi pesat, mulai dari ARPANET di tahun 1960-an hingga menjadi jaringan global seperti Internet saat ini. Seiring berjalannya waktu, teknologi jaringan juga mengalami peralihan dari jaringan kabel ke jaringan nirkabel seperti Wi-Fi dan 5G, yang menawarkan kecepatan dan akses lebih luas. Selain itu, adopsi jaringan berbasis cloud semakin memudahkan akses data dan kolaborasi jarak jauh. Inovasi lain seperti IPv6, SDN, dan IoT semakin memperluas ekosistem jaringan, menghubungkan miliaran perangkat serta mendukung berbagai industri dalam menciptakan solusi yang lebih aman dan efisien.

Keamanan jaringan menjadi semakin penting seiring dengan peningkatan kompleksitas infrastruktur serta tingginya ancaman siber, seperti peretasan dan serangan DDoS. Sistem keamanan jaringan saat ini menggunakan berbagai pendekatan berlapis, termasuk penggunaan firewall, enkripsi, dan pemantauan aktivitas jaringan secara real-time. Di sinilah pentingnya peran Intrusion Detection System (IDS), seperti Snort, yang digunakan untuk mendeteksi potensi serangan di

jaringan [1]. Snort bekerja dengan memonitor lalu lintas jaringan dan mendeteksi anomali berdasarkan aturan yang ditetapkan. Jika ada ancaman, Snort segera memberikan peringatan sehingga administrator dapat bertindak cepat untuk mencegah kerusakan lebih lanjut [1][2].

Telegram merupakan aplikasi perpesanan berbasis *cloud*, juga dapat digunakan sebagai alat notifikasi *real-time* melalui integrasi API. Integrasi Snort dengan Telegram memungkinkan administrator jaringan menerima notifikasi serangan secara cepat dan efisien, sehingga pengelolaan keamanan jaringan dapat lebih optimal, tanpa harus terus-menerus berada di depan komputer [1].

II. LANDASAN TEORI

A. Snort dan Deteksi Intrusi

Pada konteks keamanan server, web server dan FTP server sering menjadi target utama serangan hacker. Web server rentan terhadap serangan seperti SQL injection, cross-site scripting (XSS), dan serangan brute force terhadap login administrator. Sementara itu, FTP server dapat dieksploitasi melalui serangan brute force password atau penyisipan file berbahaya yang dapat digunakan untuk eskalasi hak akses. Selain itu, serangan melalui ICMP (ping) juga sering digunakan dalam reconnaissance attack, di mana penyerang mencoba mengumpulkan informasi tentang jaringan sebelum melancarkan serangan yang lebih besar. Snort dapat digunakan untuk mendeteksi pola-pola serangan tersebut dengan aturan khusus yang mampu mengenali lalu lintas mencurigakan, seperti permintaan ICMP yang tidak biasa atau upaya akses yang berulang ke FTP dan web server [2].

Pada penelitian ini, dikembangkan sebuah respon terhadap serangan secara otomatis, Snort dapat diintegrasikan dengan Python agar dapat mengambil tindakan lebih lanjut ketika mendeteksi ancaman. Salah satu implementasi yang bisa dilakukan adalah dengan menghubungkan Python ke API Telegram untuk mengirimkan pesan peringatan kepada administrator jaringan. Dengan cara ini, ketika Snort mendeteksi serangan tertentu, skrip Python dapat mengambil log kejadian, memprosesnya, dan mengirimkan peringatan langsung ke akun Telegram administrator atau tim keamanan siber. Hal ini memungkinkan deteksi dini dan respons cepat terhadap ancaman yang muncul, sehingga serangan dapat ditangani sebelum menyebabkan kerusakan yang lebih besar.

Selain deteksi dan respons otomatis, analisis tren serangan juga menjadi aspek penting dalam strategi keamanan jaringan. Salah satu metode yang dapat digunakan adalah Facebook Prophet, sebuah library berbasis Python yang dirancang untuk analisis time-series dan peramalan tren. Dengan Prophet, data serangan yang terdeteksi oleh Snort dapat dianalisis untuk mengidentifikasi pola dan anomali dalam lalu lintas jaringan. Misalnya, peningkatan mendadak dalam jumlah serangan DDoS atau percobaan login yang gagal dapat menjadi indikator adanya kampanye serangan yang sedang berlangsung. Dengan menggunakan metode ini, administrator jaringan dapat lebih proaktif dalam mengantisipasi ancaman sebelum serangan mencapai puncaknya [2].

Untuk memvisualisasikan hasil analisis ini, sebuah antarmuka berbasis terminal dapat dikembangkan untuk menampilkan tren serangan yang dianalisis dengan Facebook Prophet. Melalui terminal ini, administrator dapat melihat prediksi peningkatan ancaman berdasarkan pola sebelumnya. Integrasi antara Snort, Python, Telegram API, dan Facebook Prophet dapat menciptakan sistem keamanan yang tidak hanya mendeteksi serangan secara real-time tetapi juga mampu memprediksi tren serangan di masa mendatang, memungkinkan tindakan preventif yang lebih efektif dalam menjaga keamanan jaringan.

B. Metode Prophet

Metode yang digunakan dalam penelitian ini adalah Facebook Prophet. Metode ini dikembangkan oleh Facebook, diperkenalkan pertama kali pada tahun 2018 untuk memprediksi data deret waktu (Time Series) yang kuat dan fleksibel [3]. Tujuan utama metode ini adalah untuk mendeteksi kebiasaan *user* Facebook yang dianggap sebagai peminatan terhadap salah satu topik pada jejaring media sosial, sehingga Facebook dapat merekomendasikan beberapa topik atau person yang dapat diikuti oleh pengguna.

Metode ini bekerja dengan memodelkan data deret waktu berdasarkan tiga komponen utama yang dijelaskan dalam formula berikut:

$$y(t) = g(t) + s(t) + h(t) + \in (t) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Keterangan:

1. *Trend* g(t) : Data dari kejadian jangka panjang yang menunjukkan perubahan naik atau turun dari waktu ke waktu;
2. *Seasonal* s(t): Data dari kejadian yang rutin terjadi pada waktu tertentu. Data ini merupakan wujud dari data musiman (seasonal) yang pada waktu tertentu mengalami pengulangan.
3. *Holiday* h(t): Data dari kejadian yang dapat dianggap sebagai peningkatan pada hari tertentu. Liburan atau holiday tidak bisa diprediksi waktu kejadiannya. Jika salah satu tanggal memang diketahui sebagai hari libur, maka data yang diciptakan pada hari itu tidak dianggap sebagai hari libur, namun dianggap sebagai data musiman (seasonal). Data di hari libur adalah data yang diperoleh tanpa tanggal khusus yang menyatakan hari tersebut adalah hari libur. Salah satu contoh adalah libur pada awal puasa atau libur lebaran, maupun beberapa hari libur lain yang tidak diketahui pasti saatnya, karena tidak berdasarkan tanggal tertentu seperti hari kemerdekaan, hari natal atau hari pendidikan.
4. *Error* ϵ(t): Merupakan komponen yang menangani kesalahan atau ketidakpastian dalam prediksi model perhitungan.

Berdasarkan formulasi tersebut, penelitian ini mencoba mengembangkan metode peramalan tersebut menjadi metode untuk mendeteksi anomali (kejadian di luar kebiasaan) untuk mendeteksi peningkatan atau penurunan terjadinya serangan terhadap komputer *server*.

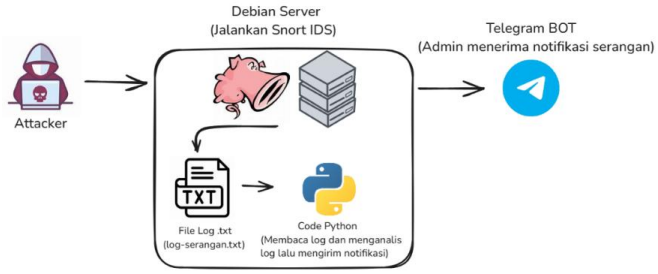
Ide pengembangan dari metode Prophet dalam penelitian ini adalah jika nilai peramalan y(t) diketahui berdasarkan jumlah serangan terkini yang telah dilaporkan oleh Snort, maka nilai h(t) yang menjadi nilai tidak diketahui, karena

menjadi efek hari libur, maka nilai tersebut adalah nilai anomali. Formulasi yang diperoleh berdasarkan penjelasan tersebut adalah sebagai berikut.

$$h(t) = y(t) - g(t) + s(t) + \in (t) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

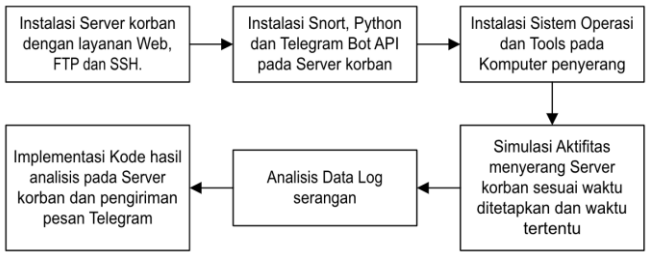
C. Metode Penelitian

Ilustrasi pada penelian ini disampaikan dalam gambar 1. Pada gambar tersebut menggambarkan bagaimana komponen-komponen perangkat lunak berinteraksi untuk mencapai tujuan utama sistem, yaitu mendeteksi serangan siber dan mengirimkan notifikasi secara *real-time* kepada administrator jaringan. Berikut ini adalah blok diagram yang digunakan seperti pada Gambar 1.



Gambar 1. Ilustrasi Kerja Sistem pada pengiriman notifikasi serangan

Metode penelitian yang dipergunakan dalam penelitian ini disampaikan dalam blok diagram pada gambar 2. Simulasi pada penelitian ini menerapkan penggunaan VirtualBox untuk komputer sebagai Server korban dan komputer penyerang. Penggunaan virtual sistem dapat mensimulasikan perangkat komputer untuk dapat menjalankan sistem operasi lain tanpa mengganggu sistem operasi utama [4][5].



Gambar 2. Blok Diagram Metode Penelitian

Pada langkah awal mempersiapkan komputer Server yang bertindak sebagai korban penyerangan. Proses instalasi komputer server disertai dengan instalasi layanan Web, FTP dan SSH. Proses dilanjutkan dengan melakukan instalasi aplikasi Snort, Python dan Telegram Bot API. Library tambahan yang dipergunakan untuk mendeteksi perubahan pada file log Snort menggunakan Watchdog. Langkah berikutnya adalah mempersiapkan komputer penyerang yang menggunakan sistem operasi Kali Linux.

Data yang dipergunakan dalam penelitian ini adalah data yang dibangun sendiri melalui bentuk serangan yang diatur secara berkala, yaitu pada waktu yang telah ditentukan untuk mewakili data *seasonal* seperti yang diminta dalam metode Prophet. Data berikutnya adalah data penyerangan yang dilaksanakan pada waktu tertentu, sebagai bentuk penggambaran data *holiday*. Kedua data tersebut berupa file log yang berisi waktu serangan dan jenis serangan pada server. Jumlah serangan dipergunakan sebagai proses analisis pada penelitian ini.

III. HASIL DAN PEMBAHASAN

A. Perangkat Penelitian

Perangkat keras (hardware) yang dipergunakan dalam penelitian ini adalah komputer dengan processor AMD Ryzen 5 5600U with Radeon Graphics 2.30 GHz, RAM DDR4 16GB, media penyimpanan menggunakan SSD 512GB. Sistem operasi yang dipergunakan adalah Windows 11 Pro.

Perangkat lunak yang dipergunakan pada komputer server adalah sistem operasi Debian versi 11. Aplikasi Web Server menggunakan Apache Web Server yang memiliki banyak pengguna karena kemudahan instalasi dan kompatibilitasnya dengan berbagai aplikasi [6]. Aplikasi FTP server menggunakan vsftpd, dan SSH server menggunakan OpenSSH. Aplikasi Snort menggunakan versi 3.0, Python versi 3.7.3. Pustaka kode program dalam bahasa pemrograman Python adalah Telegram API menggunakan Telebot dan Watchdog yang dipergunakan untuk mendeteksi perubahan file .log yang diciptakan oleh Snort.

Simulasi pada komputer server dan komputer penyerang menggunakan Oracle Virtual Box versi 7.0. Kedua mesin virtualisasi menggunakan processor single core dan memori 2 GB.

B. Pembahasan

Data yang dibuat secara mandiri melalui simulasi serangan selama 30 hari, mulai tanggal 01 Januari 2025 hingga tanggal 30 Januari 2025. Data jumlah serangan yang telah direkap oleh Python pada salah satu layanan server yaitu Server SSH dalam penelitian disampaikan pada Tabel 1.

Tabel 1. Rekap Data Log Serangan

Tanggal	Waktu	Jumlah Serangan
01/01/2025	00:00 – 12:00	92
	12.01 – 23.59	77
02/01/2025	00:00 – 12.00	88
	12.01 – 23.59	70
03/01/2025	00:00 – 12.00	89
	12.01 – 23.59	87
04/01/2025	00:00 – 12.00	77
	12.01 – 23.59	72
05/01/2025	00:00 – 12:00	88
	12.01 – 23.59	85
06/01/2025	00:00 – 12.00	83
	12.01 – 23.59	74
07/01/2025	00:00 – 12.00	85
	12.01 – 23.59	88
08/01/2025	00:00 – 12.00	73
	12.01 – 23.59	90
09/01/2025	00:00 – 12.00	82
	12.01 – 23.59	97

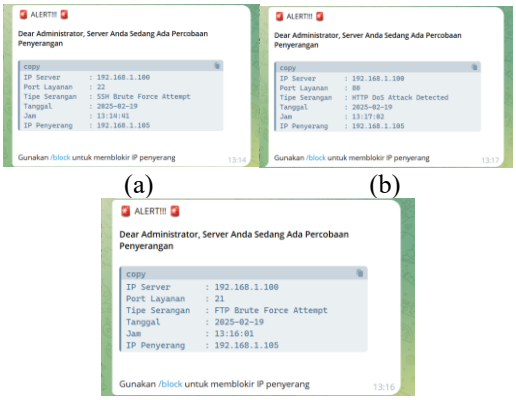
10/01/2025	00:00 – 12.00	54
	12.01 – 23.59	112
11/01/2025	00:00 – 12.00	44
	12.01 – 23.59	63
12/01/2025	00:00 – 12:00	105
	12.01 – 23.59	106
13/01/2025	00:00 – 12.00	83
	12.01 – 23.59	77
14/01/2025	00:00 – 12.00	88
	12.01 – 23.59	93
15/01/2025	00:00 – 12.00	73
	12.01 – 23.59	68
16/01/2025	00:00 – 12.00	90
	12.01 – 23.59	83
17/01/2025	00:00 – 12.00	75
	12.01 – 23.59	68
18/01/2025	00:00 – 12.00	88
	12.01 – 23.59	101
19/01/2025	00:00 – 12:00	85
	12.01 – 23.59	72
20/01/2025	00:00 – 12.00	60
	12.01 – 23.59	88
21/01/2025	00:00 – 12.00	83
	12.01 – 23.59	78
22/01/2025	00:00 – 12.00	84
	12.01 – 23.59	77
23/01/2025	00:00 – 12.00	76
	12.01 – 23.59	81
24/01/2025	00:00 – 12.00	86
	12.01 – 23.59	75
25/01/2025	00:00 – 12:00	96
	12.01 – 23.59	82
26/01/2025	00:00 – 12.00	50
	12.01 – 23.59	103
27/01/2025	00:00 – 12.00	79
	12.01 – 23.59	66
28/01/2025	00:00 – 12.00	74
	12.01 – 23.59	73
29/01/2025	00:00 – 12.00	79
	12.01 – 23.59	83
30/01/2025	00:00 – 12.00	85
	12.01 – 23.59	88

Salah satu bentuk serangan terhadap layanan Web pada komputer server diantaranya adalah DoS (Denial of Service), yaitu bentuk serangan dengan mengirimkan banyak permintaan ke sistem atau ke jaringan secara bersamaan, sehingga membebani sumber daya sistem dan menyebabkan kinerja yang buruk atau bahkan kegagalan total [7][8]. Bentuk serangan kedua yang dipergunakan sebagai simulasi dalam penelitian ini adalah *Brute Force Attack*, yaitu salah satu bentuk serangan yang bertujuan untuk mencoba segala kemungkinan kombinasi karakter untuk mencari akun yang valid [8]. Serangan ini dapat dilakukan pada layanan FTP dan SSH yang memiliki akun valid untuk menggunakannya.

Komputer server yang memantau lalu lintas jaringan menggunakan Snort IDS, selanjutnya memproses deteksi serangan berdasarkan aturan-aturan yang telah ditetapkan. Ketika Snort mendeteksi adanya serangan seperti Denial of Service (DoS) atau Brute Force Attack pada layanan SSH dan FTP, informasi deteksi tersebut dicatat dalam file Log.

Script Python yang ditulis menggunakan library Watchdog memantau perubahan pada file Log secara *real-time*. Pada saat terdapat perubahan dalam file Log, script akan memproses informasi dan mengirimkan notifikasi ke Telegram melalui Telegram Bot API menggunakan library Telebot.

Berdasarkan hasil uji coba sistem deteksi serangan pada layanan server. Aplikasi Python yang dibangun dengan integrasi pustaka kode dari Telegram dan Watchdog dapat mengirimkan pesan dengan baik.

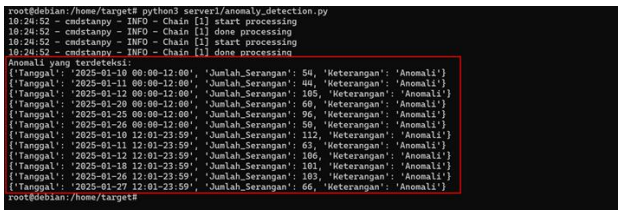


Gambar 3. Bentuk Notifikasi saat terdapat serangan pada layanan (a) SSH, (b) Web dan (c) FTP

Pada proses pengujian anomali dengan metode Prophet, diawali dengan pencarian data *trend*. Metode yang dipergunakan adalah metode Moving Average. Sedangkan pada proses pencarian data *seasonal* dapat menggunakan library statsmodels dari Python. Kode program yang dituliskan dalam Python disampaikan sebagai berikut.

```
import pandas as pd
from statsmodels.tsa.seasonal import seasonal_decompose
data = pd.read_csv('http-attack.log', parse_dates=True,
index_col='Date')
result = seasonal_decompose(data[jml_serangan], model='additive',
period=30)
```

Berdasarkan formulasi yang diusulkan, maka dapat diperoleh data *holiday* yang dikenali pada penelitian ini sebagai data anomali. Hasil deteksi anomali pada file Log oleh Python untuk menyimpan informasi dapat diamati pada data gambar berikut.



Gambar 4. Hasil uji coba penerapan Anomali Detection

Saat anomali terdeteksi, maka Snort akan diminta untuk melakukan proses blokir alamat IP penyerang. Administrator jaringan dapat melakukan pengecekan manual terhadap alamat IP tersebut menggunakan Telegram.

Gambar 5. Pengecekan manual alamat IP Penyerang

Penelitian ini juga menambahkan kode program yang dapat melakukan proses *unblock* pada alamat komputer IP penyerang melalui aplikasi Telegram, seperti disampaikan pada gambar 6 berikut.

Gambar 6. Unblock IP Penyerang menggunakan Telegram

IV. KESIMPULAN

Metode analisis anomali yang diusulkan menggunakan Prophet dari Facebook mampu meningkatkan kemampuan Snort dalam mendeteksi peningkatan serangan yang terjadi pada aplikasi layanan *server*. Snort sebagai aplikasi dengan kode terbuka dapat dimodifikasi melalui tambahan kode dengan Python untuk meningkatkan performanya dalam mendeteksi serangan dalam jaringan komputer.

REFERENSI

[1] Tommy Purnama dkk, ”Implementasi Intrusion Detection System (IDS) SNORT sebagai sistem keamanan menggunakan whatsapp dan telegram sebagai media notifikasi”, Jurnal Ilmiah Teknologi Informasi dan Komunikasi (JTIK), Vol.14 No.2 (September,2023), 359.

[2] Abdulrezzak, S., Sabir, F., “An Empirical Investigation on Snort NIDS versus Supervised Machine Learning Classifiers”, Journal of Engineering, 2023, https://doi.org/10.31026/j.eng.2023.02.11

[3] Evydian Rosa Putri dan Budhi Kristianto, “Penerapan Algoritma Prophet Facebook untuk memprediksi jumlah calon mahasiswa baru”, Jurnal Penerapan Sistem Informasi (Komputer & Manajemen), Vol.5 No.04 (Oktober,2024), 1589.

[4] Sofyan Mufti Prasetiyo dkk, “Mesin Virtual (Virtual Machine): Sekilas Tentang Tujuan, Fungsi, Keuntungan, Dan Pengelolaan Dari Mesin Virtual”, Buletin Ilmiah Ilmu Komputer dan Multimedia, Vol. 1 No. 6 (April, 2024), 746.

[5] Amarudin dan Atri Yuliansyah, “Analisis Penerapan MikroTik Router Sebagai User Manager Untuk Menciptakan Internet Sehat menggunakan Simulasi Virtual Machine”, Jurnal TAM (Technology Acceptance Model), Vol.9 No.1 (July, 2018),62.

[6] Albert Yakobus Chandra, “Analisis Performasi antara APACHE & NGINX Web Server dalam Menangani Clinet Request”, Jurnal Sistem dan Informatika (JSI), Vol. 14 No.1 (November,2019), 49.

[7] Candra Adi Winanto, “Deteksi Serangan Denial of Service Menggunakan Artificial Immune System”, Fakultas Ilmu Komputer Universitas Sriwijaya, Vol. 2 No. 1 (Desember, 2016), 456.

[8] Daryn Ramadhani Az Zahra, “Penerapan dan Pengujian Keamanan SSH Pada Server Linux menggunakan Hydra”, Journal of Internet and Software Engineering, Vol. 1 No. 3 (Juni, 2024), 4.