REPOSITOR, Vol. 4, No. 1, Februari 2022, Pp. 95-102
ISSN : 2714-7975
E-ISSN : 2716-1382

Visualisasi Data Attacker Activity Log Portable Modern Honey Network

Syaifuddin[1], Wisnu Bayu Ahadin[*2], Zamah Sari[3]
[1,2,3]Teknik Informatika, Universitas Muhammadiyah Malang
saifuddin@umm.ac.id[1], wisnu_437168@webmail.umm.ac.id[*2], zamahsari@umm.ac.id[3]

Abstrak

Internet sudah menjadi komoditas utama dalam hal komunikasipada era ini. Dengan seiring perkembangan zaman, serangan yang terjadi di internetpun semakin berkembang. Untuk mencegah sebuah serangan terjadi sudah banyak sistem atau program yang dikembangkan untuk menghalau suatu serangan, salah satu diantaranya yaitu honeypot. Honeypot merupakan aplikasi yang dikembangkan guna mampu menahan, mendeteksi, serta mencatat serangan yang masuk kedalam suatu jaringan server. Dengan menggunakan informasi yang didapat oleh honeypot akan lebih memudahkan dalam hal mencegah kerusakan yang teerjadi apabali suatu serangan terjadi di suatu saat mendatang. Dengan segala fitur yang terdapat pada honeypot tersebut, masih belum banyak pengguna ataupun peminat dari kalangan komunitas keamanan jaringan dikarenakan honeypot termasuk kedalam kategori yang cukup rumit dalam hal pengelolaan serta pemeliharaannya. Modern Honey Network atau yang biasa dikenal MHN merupakan sebuah sistem manajemen yang mampu menjalankan banyak sensor honeypot dalam waktu yang bersamaan secara singkat. MHN menggunakan sensor – sensor honeypot untuk mengumpulkan data yang berhubungan dengan serangan ke dalam jaringan. MHN menyimpan data – data serangan dari sensor – sensor yang ada dalam bentuk file log.MHN bertujuan untuk mengatasi permasalahan dari rumitnya pemeliharaan serta pengelolaan dari honeypot. Akan tetapi kekurangan dari MHN ini yaitu tidak adanya fitur visualisasi dari data – data serangan yang sudah tersimpan tadi, hal ini tentunya akan mempersulit kinerja dari seorang administrator jaringon Ketika akan melakukan Analisa terhadap suatu serangan yang masuk kedalam server jaringan mereka. Oleh karena itu dibutuhkan sebuah program yang dapat memvisualisasikan data log yang ada pada MHN. Untuk melakukan visualisasi tersebut pada penelitian ini digunakan sebuah program yang bernama Grafana guna memvisualisasikan seluruh data informasi yang berada dalam log data pada MHN. Dengan menghubungkan Grafana dengan MHN diharapkan seorang Administrator jaringan akan lebih mudah dan cepat dalam melakukan Analisa serangan yang terjadi dalam server jaringan mereka dan dapat mengatasinya dengan cepat.

Kata Kunci: *Honeypot, MHN, Real-time, Grafana*

Abstract

The internet has become a major commodity in terms of communication in this era. Along with the times, cyber attacks that occur on the internet are growing. To prevent an attack from happening, many systems or programs have been developed to ward off an attack, one of it is honeypot. Honeypot is an application developed to be able to withstand, detect, and record attacks that enter network server. By using the information obtained by the honeypot, it will be easier to prevent damage that occurs when an attack occurs in the future. With all the features contained in the honeypot, there are still not many users or enthusiasts from the network security community because honeypot are include in category that is quite complicated in terms of management and maintenances. Modern Hone Network or commonly known as MHN is a management system that is able to run multiple honeypot sensors at the same time in a short time. MHN uses honeypot sensors to collect data related to attack that happen in the network. MHN stores attack data from existing sensors in the form of log files. MHN aims to overcome problems from the complexity of maintenance and managements of the honeypot. However, the disadvantages of this MHN is that there is no visualization feature of the attack data that has been stored earlier, this will certainly complicate the performance of a network administrator when analyzing an attack that enters their server. Therefore we need a program that can visualize the existing log data on MHN. To perform the visualization in this case, a program called Grafana was

used to visualize all the information data contained in the data log on MHN. By connecting Grafana with MHN, it is hoped that a network administrator will find it easier and faster to analyze attacks that occur on their network servers and can overcome them quickly.

Keywords: *Honeypot, MHN, Real-time, Grafana*

1. Pendahuluan

Semakin berkembangnya teknologi saat ini membuat tindak kriminalitas baik secara langsung maupun tidak langsung yang memanfaatkan teknologi informasi dan komunikasi ikut berkembang. Pemanfaatan berbagai fasilitas seperti internet, perangkat digital pengolahan citra, *smartphone*, *email*, dan lainnnya menjadikan berbagai pihak lebih mudah untuk melakukan suatu tindak kejahatan. Saat ini serangan *cyber* di internet semakin mengalami peningkatan[1], serangan *cyber* biasanya menyerang server – server sebuah perusahaan yang dimana sangat dibutuhkan untuk mengakses *website*, atau *database* perusahan tersebut. Bila serangan *cyber* tersebut terus terjadi, akan berdampak buruk terhadap kinerja suatu server sehingga membuat para pengguna server tersebut merasa sangat tidak nyaman dikarenakan hal tersebut dapat menghambat pekerjaan mereka.

Berdasarkan data yang dirilis oleh Badan Siber dan Sadni Negara (BSSN), sepanjang tahun 2020 (1Januari – 30 Desember), jumlah Anomali trafik yang berhasil tercatat mencapai 495.337.202 anomali. Puncak anomaly trafik tertinggi mencapai 7.311.606 anomali pada Desember 2020.

Ada berbagai jenis dari serangan *cyber* diantaranya, *Malware, DoS / DDoS, Phising, SQL Injection* dan lainnya[2]. Serangan yang dilakukan dengan melakukan paket secara terus menerus dengan menggunakan banyak computer oleh banyak host atau biasa disebut *Distributed Denial of Service (DDoS)*[3] dapat membuat jaringan menjadi tidak stabil atau bahkan tidak dapat diakses dikarenakan server tidak dapat menangani jumlah request yang berlebihan dari *resource* yang disediakan oleh server. Serangan *Distributed Denial of Service* merupakan serangan *DoS* biasa tetapi dilakukan secara terdistribusi, maksudnya serangan ini dilakukan oleh sebuah host dengan melakukan *remote* pada komputer lain dengan jumlah banyak kemudian menyerang sebuah server. Akibatnya server yang terkena serangan tersebut akan mendapatkan *request* yang membanjiri sistem komputer pada jaringan server tersebut sehingga server tidak dapat melayani seluruh *request* dari tiap *host*. Hal ini akan terus terjadi jika serangan tersebut tidak dihentikan atau melakukan suatu tindak pencegahan dari serangan *Distributed Denial of Service (DDoS)* tersebut.

Dengan banyaknya jenis serangan yang ada, maka dari itu seorang administrator system diharapkan mampu untuk melakukan pertahanan terhadap system yang mereka miliki karena ada banyak dampak negatif yang disebabkan oleh serangan *cyber*, kemungkinan terburuknya dapat menyebabkan pencurian terhadap data – data penting berupa informasi pribadi atau data penting perusahaan, kemudian ada juga kemungkinan terjadinya sabotase terhadap sistem yang disebabkan *malware* atau *DDoS*. Oleh karena itu seorang administrator system harus mengerti bagaimana cara mempertahankan sistem mereka dari serangan *cyber* agar tidak mengganggu *user / client* untuk mengakses sistem mereka.

Salah satu tindakan dalam melakukan pencegahan yaitu dengan memperkuat pertahanan dari sistem, biasanya dengan menggunakan *Honeypot. Honeypot* adalah sebuah system yang dimana bertindak sebagai *dummy target* untuk mengumpulkan informasi – informasi mengenai serangan yang ada pada suatu sistem[4]. Cara kerja *Honeypot* mirip seperti server pada umunya tetapi tidak memberikan informasi nyata atau sebenarnya. *Honeypot* digunakan untuk mendeteksi serta menyimpan informasi dari serangan *DDoS, a*dapun informasi yang tersimpan berupa *ip* penyerang sampai asal negara dari *ip* tersebut, informasi – informasi tersebut tersimpan dalam bentuk data *log.*

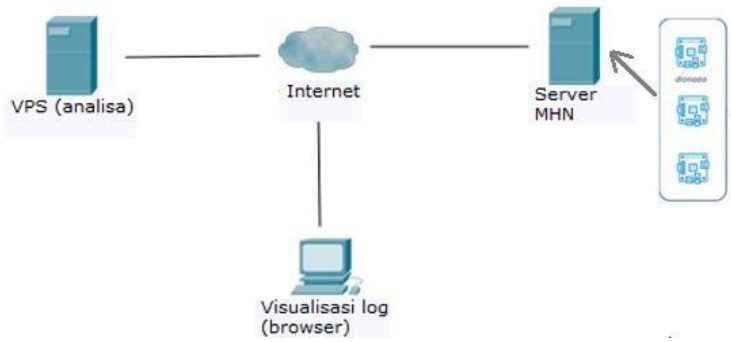
Dari data *log* tersebut dapat dilakukan analisa terhadap suatu serangan karena *log* tersebut menyimpan berbagai jenis informasi berupa *ip address* penyerang, waktu penyerangan, jenis serangan, dan juga *port* yang diserang. Karena informasi yang tersimpan dalam data log *Honeypot* sulit untuk dibaca secara langsung, oleh karena itu untuk mempermudah dalam proses pembacaan informasi yang ada dalam data *log Honeypot* tersebut kita dapat memvisualisasikannya ke dalam bentuk grafik sehingga lebih memudahkan dalam proses pembacaannya.

Ada banyak cara yang dapat di gunakan untuk memvisualisasikan data – data atau file log yang kita punya, salah satunya dengan memanfaatkan aplikasi web visualisasi, salah satu aplikasi web visualisasi yang di gunakan dalam penelitian ini yaitu dengan menggunakan *Grafana. Grafana* merupakan *multi-platform open source* yang banyak di gunakan sebagai alat bantu untuk memvisualisasika data – data[5].

Dengan pemaparan diatas, penelitian ini bertujuan untuk membantu administrator sistem untuk dapat menganalisa jaringan suatu server dengan cepat serta dapat dengan segera mengatasi masalah yang ada dengan akurat[6]. Dengan memvisualisasikan data *log* secara *real-time* administrator dengan mudah dapat melihat *port* atau servis yang terkena serangan sehingga dapat dengan segera diambil tindakan untuk masalah yang terjadi tanpa harus menunggu lagi untuk melakukan eksport data *log* yang ada pada *Honeypot*.

2. Metode Penelitian

2.1 Gambaran Sistem



Gambar 1. Gambaran Umum Sistem

Seperti terlihat pada Gambar 1, terdapat tiga komponen utama dalam system yang akan dibuat yaitu server MHN yang terhubung langsung ke internet, *VPS* yang sudah dipasang algoritma untuk melakukan Analisa, dan client untuk menampilkan hasil Analisa berupa visualisasi pada browser [9], [10]. Dari komponen – komponen diatas memiliki fungsi masing–masing, pada server *MHN* berfungsi sebagai dummy server sebagai sasaran serangan, dalam server *MHN* juga di install sensor – sensor seperti *dionaea, p0f, cowrie*, dan lainnya untuk menangkap serangan yang masuk *MHN* mencatat segala jenis akttivittas serangan dan disimpan dalam bentuk file *log*. Dari server *MHN* kemudian mengirimkan file *log* ke VPS untuk menganalisa data – data pada file *log* tersebut, data yang diterima VPS kemudian diolah. Setelah selesai melakukan pengolahan file log tadi kemudian hasilnya dapat diakses melalui browser pada computer client, adapun hasil pengolahan yang ditampilkan berupa visualisasi garfik dalam bentuk dashboard.

2.2 Pengumpulan data

Dalam proses pengumpulan data metode yang di gunakan dalam mengumpulkan data yang di gunakan dalam penelitian ini dengan memanfaatkan *Modern Honey Network (MHN)* dimana *MHN* dapat mengumpulkan data dari sensor yang telah di pasang pada *MHN*. *Honeypot* berfungsi sebagai perekam semua aktifitas jaringan dalam penelitian ini. pada saat *honeypot* di jalankan kemudian mendeteksi adanya sebuah serangan ke dalam jaringan kita, *honeypot* kemudian akan merekam semua kejadian tersebut lalu di kirimkan kepada *MHN* dengan port 1000 (*default*).

2.3 Pengambilan data

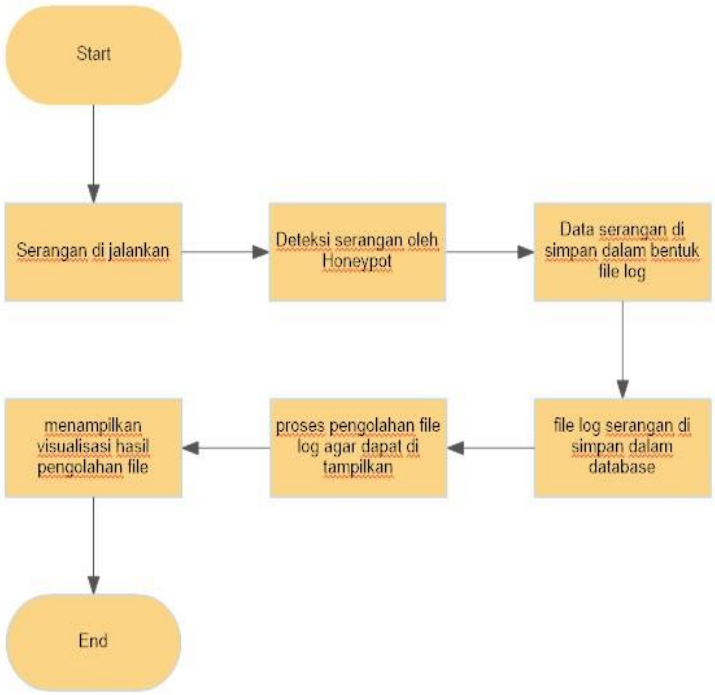
Penggunaan MHN merupakan yang paling baik dalam mengatur honeypot, termasuk dalam proses pengiriman log data, seperti pada Gambar 2. Pengumpulan data oleh sensor kemudian di lanjutkan pada proses pengambilan data yang sudah terkumpul dalam MHN. Untuk melakukan pengambilan data peneliti memanfaatkan API yang sudah tersedia sebagai fitur pada MHN, sehingga akan lebih mudah di terapkan dalam pemrograman untuk Analisa.

Gambar 2. Pengambilan Log Data

Server yang berguna untuk Analisa akan meminta data ke MHN dengan melalui API yang ada. Kemudian MHN akan mengambil data yang ada dalam database berupa file json yang kemudian diberikan sebagai bentuk respon permintaan dari MHN. File json tersebut kemudian akan di Analisa sehingga dapat di tampilkan kepada user.

2.4 Proses Pengujian

Dalam peroses pengujian akan dilakukan dengan mengikuti alur skema, mulai dari proses serangan terjadi hingga memvisualisasikan hasil dari serangan tersebut agar mudah dipahami. Detail alur skema tersebut dapat diperhatikan dalam flowchart pada Gambar 3



Gambar 3. Alur Proses Pengujian

Dalam proses pengujian Langkah awalnya akan di lakukan serangkaian serangan untuk melakukan analisa. Dengan menggunkan serangan *DoS/DDoS* yang di lakukan beberapa komputer. Untuk mengetahui sitem itu bekerja maka *honeypot* akan di jalankan agar semua serangan dapat masuk pada jebakan honeypot. Dan ketika hasil dari sebuah log sudah di tulis pada *text file*, lanjut di kirimkan dan di lakukan analisa.

Proses dalam Analisa yang terjadi pada Gambar 3 menggunakan alur serangan dari proses scanning pada ip target. Setelah proses scanning dilakukan, di dapatkan haril berupa beberapa port yang terbuka dalam ip tersebut. Setelah serangan dilakukan, kemudian honeypot akan mulai merekam atau mencatat semua kegiatan yang terjadi dalam bentuk log lalu dikirimkan ke MHN. Hasil log yang sudah tersimpan tadi dapat di ambil dengan memanfaatkan fitur API yang ada pada MHN.

3. Hasil Penelitian dan Pembahasan

Dalam bab ini akan dijelaskan mengenai serangan yang masuk ke honeypot, bagaimana mengambil data dari log yang di MHN ke Grafana dan bagaimana memvisualisasikan data log tersebut.

3.1 Implementasi Honeypot dan MHN

Dalam melakukan implementasi pada studi kasus ini digunakan Raspberry pi 3 model B+ dan Proxmox sebagai alat pendukung Honeypot. Proxmox berfungsi sebagai tempat untuk melakukan instalasi VM dari MHN. Instalasi sensor honeypot dilakukan masing–masing sensor pada raspberry dengan memanfaatkan script yang ada pada MHN.

Sensors

	Name	Hostname	IP	Honeypot	UUID	Attacks
1-	Sensor-Dionaea-dionaea	Sensor-Dionaea	10.251.30.5	dionaea	a275315a-597a-11eb-84a2-de68f42fdca2	889418
2-	Sensor-Dionaea-p0f	Sensor-Dionaea	10.251.30.5	p0f	f529b1e2-5983-11eb-84a2-de68f42fdca2	2202388
3-	Sensor-Dionaea-elastichoney	Sensor-Dionaea	10.251.30.5	elastichoney	26b7cbae-5984-11eb-84a2-de68f42fdca2	10070
4-	Sensor-all-snort	Sensor-all	10.251.30.6	snort	3bba5d0e-598a-11eb-84a2-de68f42fdca2	1996
5-	Sensor-all-cowrie	Sensor-all	10.251.30.6	cowrie	8958447c-598a-11eb-84a2-de68f42fdca2	122384

Gambar 4. Sensor yang Sudah Terpasang

Seperti terlihat pada gambar 4, ada 5 sensor yang sudah terpasang dalam MHN. Sensor–sensor tersebut sudah berjalan dengan baik guna mendapatkan data log dari serangan yang masuk kedalam honeypot.

3.2 Import Data Log Dari MHN Menunju Database di Grafana

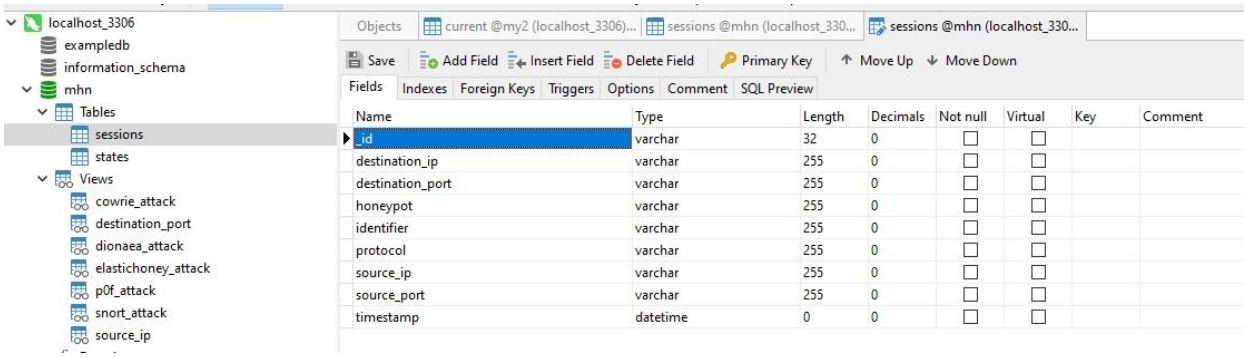
Untuk melakukan visualisasi data pada Grafana, terlebih dahulu data yang berada dalam log yang tersimpan pada MHN harus di import terlebih dahulu menuju database MySQL yang sudah dibuat pada server Grafana. Untuk memindahkan data log dibutuhkan sebuah program tambahan, dalam penelitian menggunakan program dengan bahasa javascript yang berfungsi mengambil data dari MHN dan memasukkannya ke dalam database MySQL pada server Grafana. Dalam program .js yang dibuat, hal yang harus diperhatikan dalam penulisan kode yaitu, detail database tempat tujuan pemindahan data, detailnya berupa nama dari database, user dan password. Kemudian detail dari MHN yang merupakan sumber dari data, detailnya meliputi host / dns yang digunakan untuk MHN, port yang digunakan, serta API key dari user MHN. Kemudian dalam file json tersebut terdapat fungsi yang bertugas untuk mengelompokkan data-data yang berada dalam data log untuk dipisahkan lalu dimasukkan kedalam tabel database di Grafana sesuai dengan jenis informasi datanya, ditunjukkan pada gambar 5, Gambar 6, dan Gambar 7.

```
const uuid = require('uuid');

var http = require('http');
var mysql = require('mysql');
const { exit } = require('process');
var connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '',
  database: 'mhn'
});

var options = {
  host: 'i-lab.umm.ac.id',
  port: 8080,
  path: '/api/session/?api_key=161c0967e9d4414b9c2b2101ccda2591&hours_ago=1'
};
```

Gambar 5. Fungsi untuk Database dan MHN

```
async function insert(){
  for (const response of responses.data) {
    await connection.query('INSERT IGNORE INTO sessions (_id, destination_ip, destination_port, honeypot, identifier, protocol, source_ip,
source_port, timestamp) VALUE (\'' + response._id + '\', \'' + response.destination_ip + '\', \'' + response.destination_port + '\', \'' +
response.honeypot + '\', \'' + response.identifier + '\', \'' + response.protocol + '\', \'' + response.source_ip + '\', \'' + response.source_port +
'\', \'' + response.timestamp + '\')', function (error, results, fields) {
      if (error) throw error;
      console.log(response._id);
    });
  }
}

async function counts() {
  await connection.query('SELECT COUNT(*) as total FROM sessions', function (error, results, fields) {
    if (error) throw error;
    var created = new Date();
    const message = `Pooling with ${responses.meta.size} and writen ${results[0].total} data, finish at: ${created}`
    connection.query('INSERT INTO states (_id, createAt, message) VALUE (' + connection.escape(uuid.v4()) + ', ' + connection.escape(created) + ', '
+ connection.escape(message) + ')', function (error, results, fields) {
      if (error) throw error;
      console.log(message);
    });
  });
}
```

Gambar 6. Fungsi Mengelompokkan Data Dari Data Log



Gambar 7. Hasil Import Data dari MHN ke Grafana

3.3 Visualisasi Data Dengan Grafana

Langkah terakhir dalam studi kasus ini yaitu bagaimana melakukan kustomisasi untuk memvisualisasikan data pada database Grafana hingga data – data serangan mudah dibaca.

Setelah berhasil memindahkan data file json dari server MHN ke database server Grafana, kustomisasi dashboard sudah bisa dilakukan. Pada dashboard Grafana yang sudah terpasang sebelumnya terdapat menu edit untuk melakukan kustom. Dalam menu edit ini digunakan untuk mnyesuaikan visualisasi, proses menampilkan data menggunakan query MySQL. Berikut merupakan hasil dari visualisasi dashboard dengan menggunakan Grafana dimana sudah berhasil menampilkan data–data detail dari serangan yang masuk kedalam server honeypot, seperti pada Gmabar 8.

Gambar 8. Hasil Visualisasi Grafana

4. Kesimpulan

Dalam laporan penelitian ini seluruh proses implementasi yang sudah dilakukan dengan melakukan uji coba sistem menunjukkan hasil yang sesuai dan dapat memvisualisasikan data serangan yang masuk pada honeypot. Hasil visualisasi dari serangan yang terjadi memberikan visualisasi data secara real-time, sehingga dapat disimpulkan sebagai berikut :

a. Honeypot dapat mendeteksi serangan yang masuk ke dalam jaringan server.
b. Hasil pengujian memberikan hasil data seara real-time.
c. Dari hasil Analisa data log, data penyerangan dikelompokkan ke dalam ip penyerang, ip yang diserang, serta port yang diserang.
d. Data log dari seerangan yang sudah tersimpan berhasil di visualisasikan dalam bentuk dashboard Grafana.

Berdasarkan hasil pengujian yang telah dilakukan dalam penelitian ini masih memiliki kekurangan serta kelemahan, saran dari penulis untuk melakukan pengembangan pada penelitian ini sebagai berikut :

a. Visualisasi data log serangan dapat dibuatkan dalam bentuk map peta, sehingga asal serangan dapat ditampilkan kedalam bentuk yang lebih detail dan menarik.
b. Setiap sensor yang dipasang dapat dibuatkan visualisasi dashboard tersendiri, sehingga data log serangan dari setiap sensor memiliki dashboard masing-masing.

Referensi

[1] R. Adrian and N. Isnianto, “Analisa Pengaruh Variasi Serangan Ddos Pada Performa Router,” no. October, pp. 1257–1259, 2016.
[2] P. D. Ali and T. Gireesh Kumar, “Malware capturing and detection in dionaea honeypot,” *2017 Innov. Power Adv. Comput. Technol. i-PACT 2017*, vol. 2017-Janua, pp. 1–5, 2018.
[3] B. Mardiyanto, T. Indriyani, I. M. Suartana, and K. Kunci, “Analisis Dan Implementasi Honeypot Dalam Mendeteksi Serangan Distributed Denial-Of-Services (DDOS) Pada Jaringan Wireless,” *32 Integer J.*, vol. 1, no. 2, pp. 32–42, 2016.
[4] S. Husnan, “Implementasi honeypot untuk meningkatkan sistem keamanan server dari aktivitas serangan,” 2013.
[5] F. Viola *et al.*, “Monitoring and Analytics at INFN Tier-1 : the next step,” vol. 07008, pp. 1–7, 2020.
[6] J. T. Informasi, A. Hariyanto, J. T. Informasi, P. Negeri, J. Jalan, and M. Po, “Peningkatana Keamanan jaringna terhadap serangan malware menggunakan teknik honyepot dionaea,” *Peningkatana Keamanan jaringna terhadap serangan malware menggunakan Tek. honyepot dionaea*, vol. 03, no. 01, pp. 1–4, 2016.
[7] I. Laksana and N. R. Rosyid, “Implementasi Honeypot Sebagai Pemantau Parameter Pada HTTP Request Untuk Mengetahui Tujuan Serangan,” pp. 364–369, 2017.
[8] D. S. Hermawan and D. Risqiwati, “Analisa Real-Time Data Log Honeypot menggunakan Algoritma K-Means pada Serangan Distributed Denial of Service,” vol. 2, no. 5, pp. 541–552, 2020.
[9] A. Wegrzynek and G. Vino, “The evolution of the ALICE O 2 monitoring system,” vol. 01042, 2020.
[10] L. Structures, M. Pieš, and R. Hájovský, “Wireless Measuring System for Monitoring the,” 2020.

ISSN: 2714-7975; E-ISSN: 2716-1382