REPOSITOR, Vol. 2, No. 5, Mei 2020, Pp. 541-552
ISSN : 2714-7975
E-ISSN : 2716-1382

Analisa Real-Time Data Log Honeypot menggunakan Algoritma K-Means pada Serangan Distributed Denial of Service

Denni Septian Hermawan*[1], Syaifuddin[2], Diah Risqiwati[3]
[1,2,3]Teknik Informatika/Universitas Muhammadiyah Malang
denni_437160@webmail.umm.ac.id[*1],saifuddin@umm.ac.id[2],diah.risqiwati@umm.ac.id[3]

Abstrak

Jaringan internet yang saat ini di gunakan untuk penyimpanan data atau halaman informasi pada website menjadi rentan terhadap serangan, untuk meninkatkan keamanan website dan jaringannya, di butuhkan honeypot yang mampu menangkap serangan yang di lakukan pada jaringan lokal dan internet. Untuk memudahkan administrator mengatasi serangan digunakanlah pengelompokan serangan dengan metode K-Means untuk mengambil ip penyerang. Pembagian kelompok pada titik cluster akan menghasilkan output ip penyerang.serangan di ambil sercara realtime dari log yang di miliki honeypot dengan memanfaatkan MHN.

Kata Kunci: *Honeypot, K-Means, Real Time, MHN*

Abstract

The number of internet networks used for data storage or information pages on the website is vulnerable to attacks, to secure the security of their websites and networks, requiring honeypots that are capable of capturing attacks on local networks and the internet. To make it easier for administrators to tackle attacks in the use of attacking groupings with the K-Means method to retrieve the attacker ip. Group divisions at the cluster point will generate the ip output of the attacker. The strike is taken as realtime from the logs that have honeypot by utilizing the MHN.

Keywords: *Honeypot, K-Means, Real Time, MHN*

1. Pendahuluan

Serangan *cyber* di internet semakin mengalami peningkatan [1]. Bermacam macam bentuk serangan *cyber* akan membuat sebuah gangguan pada *system* yang akan menyebabkan server tersebut tidak bisa di gunakan, tidak dapat melayani permintaan pelanggan, dan yang lainnya. Target penyerangan adalah sistem layanan daring yang mencapai 29,3 % dari kebanyakan target [2]. Serangan *cyber* umumnya menyerang server–server perusahaan yang dibutuhkan oleh pengguna untuk mengakses *website*, *system*, atau *database* mereka. Jika serangan tersebut terus terjadi maka akan memperburuk suatu server, dikarenakan server yang sedang berjalan tersebut tidak dapat menjalakan operasional seperti seharusnya dan berdampak juga pada kenyamanan para pengguna server.

Serangan yang melakukan pengiriman paket secara terus menerus dengan banyak komptuter seperti *Distributed Denial Of service (DDoS)* yang di lakukan oleh banyak *host* [3] akan membuat jaringan menjadi tidak stabil, bahkan bisa saja jaringan tersebut sampai tidak bisa di pakai karena banyak sekali request hingga memenuhi *resource* dari server. Serangan *Distributed Denial of Service (DDoS)* berasal dari serangan *DoS* biasa yang berjumlah satu sumber penyerang an terkadang serangan menggunakan serangan yang terdistribusi, dimana serangan ini di lakukan oleh sebuah *host* yang melakukan *remote* pada komputer lain (*Zombie*) dengan jumlah yang banyak kemudian menyerang sebuah *server* [4]. Maka komputer server yang terserang akan mendapati *request* yang membanjiri [5] suatu sistem komputer pada jaringan server tersebut dan server tidak mampu melayani semua *request* dari tiap *host*. Hal ini tidak akan berhenti jika serangan tersebut tidak kunjung di hentikan, atau memasang pengaman untuk serangan *Distributed Denial of Service (DDoS)* tersebut.

Sebagian besar pada sebuah keamanan pada suatu sistem memfokuskan pada pertahanan, untuk metode pertahanan suatu sistem server dalam menggunakan Honeypot [6]. Honeypot merupakan sebuah sistem mampu menjadikan server menjadi umpan yang di gunakan untuk mengumpulkan informasi tentang penyerang pada sebuah sistem. Honeypot bekerja selayaknya seperti sebuah server namun tidak memberikan informasi seperti server yang nyata

atau server sebenarnya yang berjalan [7]. Honeypot dapat di gunakan untuk mendeteksi dan menyimpan informasi dari serangan DDoS sampai pada pendeteksi asal negara [8] ataupun ip yang menyerang server kemudian memberikan hasil serangan berupa data log.

Dari data log tersebut dapat di gunakan untuk menganalisis sebuah serangan karena log tersebut menyimpan data penyerang mulai dari ip, port yang di serang dan waktu yang di serang. Dengan menggunakan data log dalam penelitian ini dapat di gunakan untuk menampilkan sebuah serangan, serta juga dapat memvisualisasikan dalam bentuk grafik. K-Means merupakan algoritma yang dapat mengelompokkan data pada tiap cluster yang dapat di gunakan untuk penelitian ini. K-Means yang merupakan data clustering non-hirarki dapat membuat suatu data berada pada kelompok atau cluster. Dengan memanfaatkan pengelompokan dari algoritma K-Means maka akan di dapatkan kelompok dengan kemiripan karakteristik. Algoritma K-Means Clustering dapat mengelompokkan data log dan mengelompokkan pada suatu cluster data yang memiliki kesamaan karakteristik waktu dari sebuah log dari Honeypot dan banyaknya serangan pada jaringan, dengan keluaran berupa tiga cluster serangan. Dari serangan tersebut dapat di ambil hasil ip penyerang terbanyak dari setiap cluster. Dengan begitu hasil dari clustering akan mempermudah mengetahui serangan DDoS yang menyerang service atau port pada server.

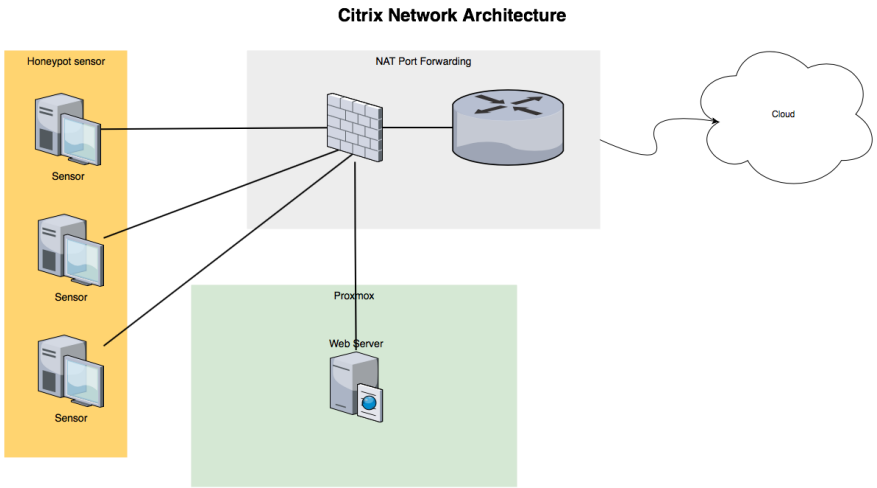
Pada penelitian sebelumnya K-Mean clustering di gunakan untuk Profiling serangan DDoS [9] yang di lakukan dari ekstraksi data dari log honeypot. Penelitian tersebut mengambil data dari sebuah log yang kemudian di lakukan perubahan format menjadi bentuk format file csv. Penelitian tersebut akan di kembangkan dengan analisa serangan yang di rekam honeypot tersebut akan di lakukan proses analisa secara realtime dari node sensor honeypot yang kemudian di kirimkan ke server utama yang di gunakan untuk analisa dari hasil log kemudian dapat di gunakan untuk memvisualisasikan hasil dari pengolahan datanya.

Hasil ahir yang di dapatkan dari proses clustering berupa IP address dari tiap penyerang dengan di kelompokkan menjadi tiga cluster. Masing masing cluster memiliki kelompok dengan ip penyerang tertinggi. Dari tiap cluster tersebut juga di ambil hasil port yang terserang paling banyak.

Dengan begitu hasil dari penelitian ini akan menghasilkan keluaran kelompok serangan cluster dan ip penyerang yang dapat membantu para administrator untuk menganalisa suatu jaringan servernya dan segera mengatasi jika terdapat serangan yang datang membuat server tidak bisa di akses oleh user karena resource telah terserang DDoS. Jika administrator dapat melihat service atau port apa yang di serang maka dapat dengan mudah seorang administrator dapat menganalisa dan mengambil tindakan tegas tanpa menunggu mengeksport terlebih dulu data log yang ada pada honeypot. Selain mengetahui serangan ip, hasil dapat memberikan informasi kepada administrator untuk mengetahui dari mana serangan tersebut berasal, berapa banyak total serangan pada waktu sedang terjadinya serangan pada server. Penelitian ini dapat memudahkan administrator dalam menemukan ip penyerang berdasarkan dari hasil clustering.

2. Metode Penelitian

2.1 Gambaran Umum Sistem



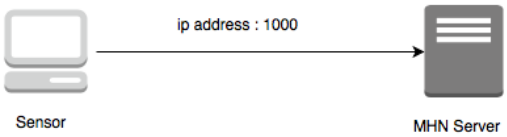
Gambar 1. Gambaran Topologi Sistem

Dalam Gambar 1 di atas terdapat tiga buah sensor, satu server proxmox yang terdapat VM sebagai web server dan juga MHN server di dalamnya, satu buah router mikrotik yang di konfigurasi menggunakan port forwarding untuk mengarahkan serangan yang datang ke arah honeypot. Dalam topologi yang ada tiga buah honeypot memiliki fungsi masing masing yang berbeda. Jenis jenis honeypot yang terpasang pada topologi ada adalah Dionaea untuk deteksi serangan DoS atau DDoS, cowrie untuk ssh bruteforce dan shell interaction, dan juga terdapat satu honeypot p0f.

Forwarding dalam mikrotik di lakukan dengan melewatkan port 21, 69, 80, 445, dan 145 ke sensor Dionaea. Untuk beberapa port tersebut secara standar berjalan di Dionaea, namun jika ingin mendapatkan serangan seperti port scanning dapat di lakukan denga nip forward ke sensor Dionaea. Untuk cowrie menggunakan port 22. Namun cowrie disini tidak di gunakan data log nya karena hanya Dionaea saja yang di gunakan dan di lakukan proses clustering pada log data tersebut.

2.2 Proses Deteksi Serangan dan penyimpanan log

Proses dari deteksi serangan yang di lakukan oleh honeypot akan di lakukan secara otomatis ketika honeypot menerima serangan. ketika honeypot terkena serangan honeypot akan mengirimkan log ke MHN server. Pengiriman yang dilakukan oleh sensor menggunakan ip address dan juga port(1000). Port yang digunakan untuk pengiriman data di atur secara otomatis ketika melakukan instalasi MHN server, seperti pada Gambar 2.

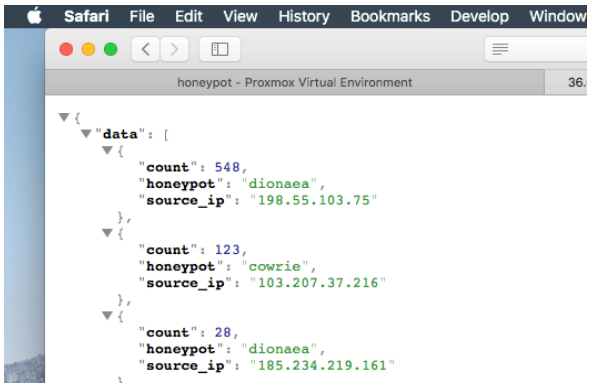


Gambar 2. Pengiriman Log ke MHN Server

Data yang telah dikirim akan di simpan oleh MHN pada database MongoDb. Untuk mendapatkan data serangan yang telah di simpan oleh MHN dengan menggunakan API yang telah disediakan oleh MHN, tanpa menggunakan database MongoDb yang digunakan oleh MHN.

2.3 Pengumpulan data

Dari hasil log yang telah di simpan dalam MHN Server yang akan digunakan untuk clustering dengan metode K-Means perlu di request terlebih dahlu. Pada sub bab 2.2 telah di jelaskan pengambilan atau akses datalog yang telah di simpan dalam database dapat di ambil dengan memanfaatkan API.

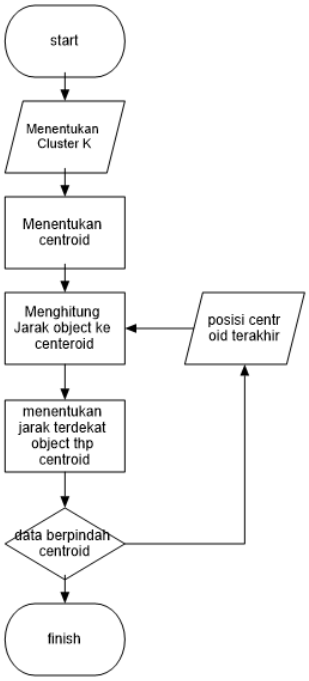


Gambar 3. Log honeypot dengan API

Pada Gambar 3 merupakan proses pemanggilan dari API MHN datalog yang di simpan pada server MHN. Dalam pemrograman data response yang berasal dari API tersebut dapat di panggil dengan menggunakan library curl. Hasil dari pengumpulan data log dari server MHN yang di ambil melalui api di lakukan pegelompokan berdasarkan waktu terlebih dahulu, karena data yang di dapat merupakan semua rekaman dari serangan tanpa ada proses perhitungan serangan.

2.4 Penerapan K-Means dalam data

Algoritma K-Means digunakan untuk mengelompokkan serangan DDoS yang di simpan dalam MHN berasal dari sensor. Data yang telah di ambil dan di hitung berdasarkan waktu serangan yang di bahas pada sub bab sebelumnya akan dilakukan penyentuan cluster dengan Algoritma K-Means. Object yang dilakukan clustering adalah hasil waktu serang di gunakan sebagai x dan total serangan dalam waktu tersebut di gunakan sebagai variable y. Dengan total centroid berjumlah 3. Diterapkannya algoritma K-Means clustering ini akan dihasilkan sebuah kelompok serangan yang memiliki kesamaan jenis serangan. serangan yang memiliki tingkatan rendah akan di kelompokkan dengan tingkatan yang rendah, dan juga untuk serangan yang tinggi akan di kelompokkan dalam kelompok yang sama tinggi. Dari hasil clustering yang terbentuk akan di ambil data tiap cluster tersebut dan di ambil waktu serangannya tiap cluster. Selanjutnya pada Gambar 4, pengambilan IP serangan diambil berdasarkan kelompok tiap cluster.



Gambar 4. Algoritma K-Means

Terdapat tahapan proses clustering dilakukan menggunakan Algoritma K-Means [10]. Berikut merupakan proses clustering terhadap data serangan yang telah terdeteksi oleh honeypot dan yang kemudian di simpan dalam database MHN.

1. Inisialisasi: menentukan nilai K sebagai centroid dan matrik ketidakmiripan (jarak) yang diinginkan. Jika perlu, tetapkan ambang batas perubahan objektif dan ambang batas perubahan posisi tersebut, di ambil K1, K2, K3, untuk centeroid.
2. Bangkitkan centroid awal secara random dari objek – objek yang tersedia sebanyak k cluster, lalu menghitung centroid dengan menggunakan Persamaan 1.

$$v = \sum_{\frac{i=1}{n}}^{n} x_1 : i = 1,2, \dots \dots \dots . \quad (1)$$

3. Hitung jarak setiap objek ke masing – masing dari masing centroid. Untuk menghitung jarak antara objek dengan centroid menggunakan euclidean distance dengan Persamaan 2.

$$\sqrt{\sum_{i=1}^{n} (xi - yi)^2 : i = 1,2, \dots \dots \dots} \quad (2)$$

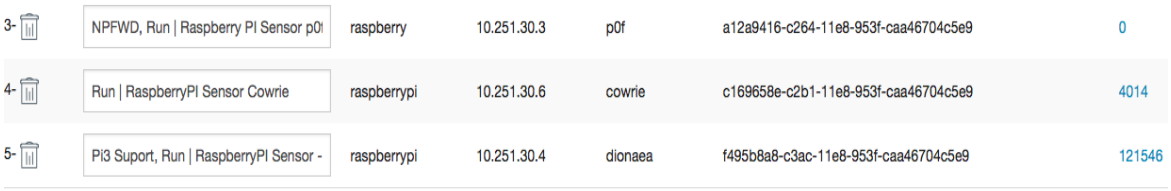
4. Mengulangi step 2 dan step 4 hingga ambang batas, atau data tidak berpindah posisi centeroid lagi.

3. Hasil Penelitian dan Pembahasan

Pada bab ini akan dijelaskan tentang implementasi dari proses Pehitungan algoritma K-Means, deteksi serangan yang masuk ke honeypot dan melakukan pengambilan IP dari tiap cluster.

3.1 Implementasi MHN dan Honeypot

Dalam implementasi ini digunakan Proxmox dan juga raspberry pi untuk honeypot. Proxmox digunakan untuk instalasi VM dari MHN installasi mhn. Instalasi untuk sesnsor honeypot dilakukan masing masing sensor dengan memanfaatkan script pada MHN.



Gambar 5 Sensor dan Total Serangan

Tiga buah sensor pada Gambar 5, diterapkan untuk mendapatkan log dari serangan yang masuk pada honeypot.

3.2 Pengumpulan data

Data yang telah di dapatkan dari serangan honeypot dikumpulkan dan di kelompokkan dalam waktu satuan jam. Berikut Tabel 1 dari data yang di kelompokkan tiap waktu satuan jam.

Tabel 1. Data Serangan Dari Satu Hari

Waktu serang	Total Serangan
0	73
1	74
2	134
3	435
4	178
5	129
6	115
7	118
8	145
9	122
10	77
11	57
12	120
13	184
14	136
15	54
16	40
17	58
18	845
19	49
20	29
21	32
22	18
23	14

3.3 Implementasi K-Means Clustering

Dari data yang telah kelompokkan dan di hitung jumlah serangan dalam satuan waktu serangan selanjutnya dapat di lakukan perhitungan dengan menggunakan K-Means clustering.

a. Menentukan centroid pertama secara random.

Dalam Tabel 2 ini dilakukan penetapan titik pusan centroid awal secara random. Dari waktu serang 0 sampai 23.

Tabel 2. Titik Pusat Centroid Awal

Titik Pusat Awal	Waktu serang	Total serangan
1	1	0
2	11	0
3	23	0

b. Perhitungan jarak object

Dari titik pusat awal yang telah di devinisikan di tetapkan titik awal serangan. dimana objek yang berjumlah 24 di hitung pada masing masing titk pusat dari centroid. Dengan menggunakan Persamaan 2 didapatkan hasil seperti berikut ini.

Jarak data pertama terhadap centroid 1: $d_{11} = \sqrt{(1-0)^2 + (0-73)^2} = 73{,}00$

Jarak data pertama terhadap centroid 2: $d_{12} = \sqrt{(11-0)^2 + (0-73)^2} = 78{,}82$

Jarak data pertama terhadap centroid 3: $d_{13} = \sqrt{(23-0)^2 + (0-73)^2} = 73{,}53$

Dari contoh satu data di atas mewakili semua data dari 24 objek yang ada, perhitungan di lakukan semua dan menghasilkan Tabel 3 perhitungan di bawah ini.

Tabel 3. Hasil Perhitungan K-Means

Waktu Serang	Signature Priority	Jarak Kelompok 1	Jarak Kelompok 2	Jarak Kelompok 3	Kelompok terdekat
0	73	73,00	73,82	76,53	1
1	74	74,00	74,67	77,20	1
2	134	134,00	133,30	134,67	1
3	435	435	435,07	435,45	1
4	178	178,20	179,37	179,01	1
5	129	120,06	120,33	121,34	1
6	115	115,10	115,10	116,24	1
7	118	118,15	118,06	119,07	2
8	145	145,16	143,03	145,77	2
9	122	122,26	122,01	122,80	2
10	77	77,52	77,00	78,08	2
11	57	57,87	57,00	58,08	2
12	120	120,50	120,00	120,50	2
13	184	184,39	184,01	184,27	2
14	136	136,61	126,03	136,29	2
15	54	55,78	54,14	54,50	2
16	40	40,72	40,31	40,60	2
17	58	60,16	68,30	58,30	3
18	845	854,17	845,02	845,01	3
19	49	52,20	49,64	49,16	3
20	29	34,66	30,52	29,15	3
21	32	37,73	33,52	33,06	3
22	18	27,65	21,09	18,02	3
23	14	26,07	18,43	14,00	3

Pada iterasi kedua digunakan titik pusat yang baru berdasarkan jumlah rata-rata data dengan menggunakan Persamaan 1 keanggotaan pada tiap kelompok dihasilkan seperti berikut.

Titik pusat kelompok 1 yang baru:

$$x: \frac{0+1+4+15}{4} = 5$$
$$y: \frac{5+11+43+0}{4} = 14.75$$

Titik pusat kelompok 2 yang baru:

$$x: \frac{2+3+5+6+7+8+9+10+11+12+13+14}{12} = 7.416$$
$$y: \frac{20+11+32+18+21+11+20+12+20+15+23+19}{12} = 18.5$$

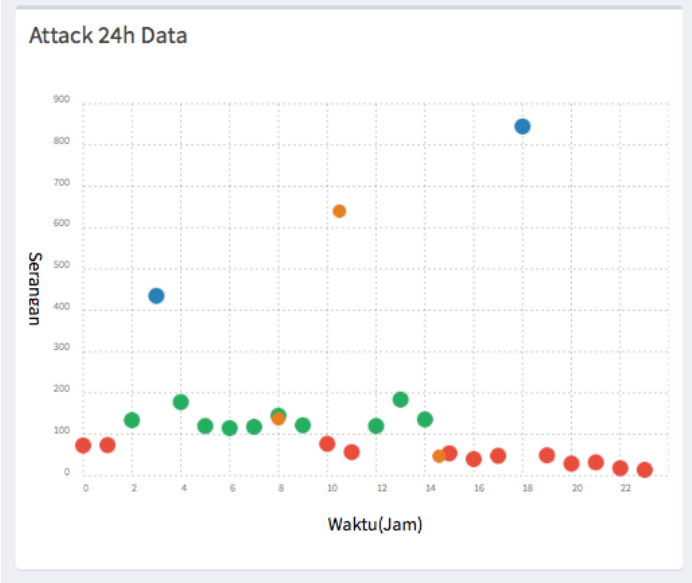
Titik pusat kelompok 3 yang baru:

$$x: \frac{21+22+23}{3} = 22$$
$$y: \frac{54+14+22}{3} = 30$$

Setelah terbentuk titik pusat yang baru maka tiap data akan dihitung kembali dengan titik pusat yang baru. Iterasi berhenti pada iterasi ke 6 karena pada iterasi tersebut tidak ada data yang berpindah lagi. Pada penilitian ini hanya diambil centroid atau titik pusat terakhir untuk digunakan sebagai hasil terhadap data serangan. Centroid atau titik pusat terakhir dapat dilihat pada Tabel 4.

Tabel 4. Titik Pusat Akhir

Titik Pusat Terakhir	Waktu serang	Total Serangan
1	10	640
2	14	47
3	8	137



Gambar 5. Serangan dalam 24 Jam

Dalam chart Gambar 5 diatas dihasilkan sebuah data object yang memiliki variable cluster. Variable tersebut digunakan untuk kelompok IP Penyerang dalam setiap Cluster Tabel 5 berikut.

Tabel 5. Kelompok Cluster

Waktu serang	Kelompok cluster
0	2
1	2
2	3
3	1
4	3
5	3
6	3
7	3
8	3
9	3
10	2
11	2
12	3
13	3
14	3
15	2
16	2
17	2
18	1
19	2
20	2
21	2
22	2
23	2

3.4 IP penyerang dari tiap cluster

Pada pengujian sistem, serangan merupakan serangan real-time yang telah ditangkap oleh honeypot. Dalam pengujian ini di ambil data real-time selama 1 hari penuh. Dengan memanfaatkan hasil dari data terahir clustering maka di dapatkan serangan dengan total dan ip sebagai berikut:

a. Cluster 1

Pada cluster petama ini penyerang melakukan serangan dengan serangan tertinggi pada jam 6 sore hari dan jam 3 pagi. Serangan ini hanya menghasilkan 2 objek, namun serangan tertinggi dari serangan pada kelompok lain.

Tabel 6. Kelompok Cluster

Jam ke	Jumlah serangan
18:00 - 18:59	845
3:00 - 3:59	435

Diambil dari Tabel 6 waktu penyerang, ip penyerang yang di dapatkan dari jam di atas dapat ditampilkan seperti Tabel 7 berikut.

Tabel 7. Kelompok Cluster

IP penyerang	Total serangan
210.124.164.133	806
115.239.248.222	272
27.79.161.195	7
1.54.211.143	4
101.164.32.167	4
…	…
74.43.56.172	1

b. Cluster 2

Pada cluster kedua ini penyerang melakukan serangan dengan serangan tertinggi pada jam 10 pagi hari dan serangan terendah di lakukan pada jam 11 malam hari. Serangan ini hanya menghasilkan 12 objek, serangan ini berada pada tingkat paling bawah.

Tabel 8. Kelompok Cluster

Jam ke	Total serangan
10	77
Jam ke	Total serangan
1	74
11	57
15	54
19	49
17	48
16	40
0	37
22	36
21	32
0	29
23	14

Diambil dari Tabel 8 waktu penyerang, ip penyerang yang di dapatkan dari jam di atas dapat ditampilkan seperti Tabel 9 berikut.

Tabel 9. Kelompok Cluster

IP Penyerang	Total serangan
103.9.79.113	8
113.160.241.158	8
14.247.52.252	8
58.186.172.157	8
110.36.226.162	7
…	…
91.98.39.21	1

c. Cluster 3

Pada cluster ketoga ini penyerang melakukan serangan dengan serangan tertinggi pada jam 1 siang hari dan serangan terendahnya terdapat pada jam 4 pagi. Serangan ini hanya menghasilkan 10 objek dalam satu cluster, serangan ini merupakan serangan dengan posisi tengah.

Tabel 10. Kelompok Cluster

Jam ke	Total Serangan
13	184
4	178
8	145
14	136
2	134
9	122
5	120
12	120
7	118
Jam ke	Total Serangan
6	115
13	184
4	178

Diambil dari Tabel 10 waktu penyerang, ip penyerang yang di dapatkan dari jam di atas dapat ditampilkan seperti Tabel 11 berikut.

Tabel 11. Kelompok Cluster

IP Penyerang	Total serangan
209.94.198.4	164
115.239.248.222	80
202.52.40.12	46
113.161.116.128	16
113.161.177.85	13
…	…
61.163.5.15	1

3.5 Hasil pengujian sistem

Dari hasil serangan yang menyerang ke honeypot akan mendapatkan log serangan dengan waktu penyerang tertinggi dan IP penyerang tertinggi. Dari table sebelumnya di menghasilkan serangan yang dapat di kelompokkan menjadi 3 centeroid dan diambil serangan dimana serangan tersebut merupakan kelompok tertinggi dalam kelompok cluster tersebut. Kelompok cluster memiliki serangan yang berbeda tiap masing-masing cluster.

Tabel 12. Tabel Centroid

No.	Centeroid	Serangan tertinggi	Total dalam kelompok	Waktu serangan serangan tertinggi
1	1	Tinggi	2	Jam 18
2	2	-	12	Jam 10
3	3	-	10	Jam 13

Dari Tabel 12 di atas diambil serangan dari tiap centeroid dengan satu object data yang serangannya tertinggi. Serangan tertinggi terletak pada centeroid ke 1 dengan total object dalam kelompok berjumlah 2 dan waktu serangan tertinggi terletak pada jam ke 18, seperti pada Tabel 13.

Tabel 13. Cluster Serangan Ip Tertinggi

No.	Centeroid	Total Serangan	IP tertinggi dalam serangan
1.	1	806	210.124.164.133
2.	2	8	103.9.79.113
3.	3	164	209.94.198.4

Dari total serangan yang ada tiap cluster dijumlahkan serangan tiap jam dari masing-masing objek tersebut menjadi satu. Di hasilkan serangan yang banyak dalam tiap cluster. Dan perlu diperhatikan bahwa serangan ini bukan merupakan serangan terbanyak dalam satuan waktu perjam namun total dari seluruh serangan tiap cluster, seperti pada Tabel 14 berikut.

Tabel 14. Total Serangan Tiap Cluster

No.	Centeroid	Total Serangan
1.	1	1280
2.	2	542
3.	3	1372

4. Kesimpulan

Dalam laporan penelitian ini proses implementasi yang di lakukan dengan uji coba sistem menunjukkan hasil yang sesuai dan dapat memberikan informasi tentang serangan yang datang dan masuk pada sensor. Hasil analisa yang di lakukan pada serangan memberikan visualisasi dihari tersebut, sehingga dapat di Tarik kesimpulan sebagai berikut:

1. Honeypot dapat mendeteksi serangan pada hari yang sudah di tetapkan di proses tersebut.
2. Proses K-Means dapat mengelompokkan serangan dengan kelompok yang sama dan memiliki kemaan karateristik tinggi rendahnya serangan yang masuk.

3. Dari hasil proses K-Means di dapatkan data penyerang dari kelompok penyerang, IP penyerang perkelompok, dan total penyerangan
4. Hasil pengujian memberikan Real Time data ketika masuk serangan yang baru serangan tersebut dapat dianalisa kembali.

Merujuk dari hasil pengujian yang telah di lakukan pada penelitian ini memiliki kekurangan dan juga kelemahan, saran dari penulis untuk mengambangkan penelitian ini sebagai berikut:

1. Menambahkan honeypot dan clusterisasi berdasarkan honeypot yang terdapat pada MHN agar serangan lebih beragam.
2. Lamanya proses clustering dapat di lakukan pada komputer yang berspefikasi lebih tinggi dan tingkat prosesing data lebih cepat.
3. Menambahkan fitur notifikasi agar sistem ini dapat memberikan informasi lagi untuk administrator ketika terdapat serangan yang masuk.

Daftar notasi

d: jarak eucledian
x: titik panjang object
y: titik tinggi object

Referensi

[1] R. Adrian and N. Isnianto, “Analisa Pengaruh Variasi Serangan DDoS Pada Performa Router,” *Pros. Semin. Nas. Teknol. Terap. SV UGM*, pp. 1257–1259, 2016.
[2] A. Hariyanto and Surateno, “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.
[3] J. Hidayat, *CEH (Certified Ethical Hacker) 500% Illegal*. Yogyakarta: Jasakom, 2004.
[4] R. V. Deshmukh and K. K. Devadkar, “Understanding DDoS Attack & its Effect in Cloud Environment,” *Procedia Comput. Sci.*, vol. 49, pp. 202–210, 2015.
[5] C. Technology, “Impact of DDOS Attacks on Cloud Environment on Application Layer,” *Proc.*, vol. 2, no. 7, pp. 362–365, 2013.
[6] S. Mahajan, A. M. Adagale, and C. Sahare, “Intrusion Detection System Using Raspberry PI Honeypot in Network Security,” *Int. J. Sci. Eng. Res. IJESC*, vol. 6, no. 3, pp. 2792–2795, 2016.
[7] 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.
[8] V. Visoottiviseth, U. Jaralrungroj, E. Phoomrungraungsuk, and P. Kultanon, “Distributed Honeypot log management and visualization of attacker geographical distribution,” *Proc. 2011 8th Int. Jt. Conf. Comput. Sci. Softw. Eng. JCSSE 2011*, pp. 23–28, 2011.
[9] Wulandari W. A, “Penggunaan Algoritma K–Means pada Datalog Honeynet untuk Mengetahui Cyber Profiling Serangan DDoS,” *J. Inforkatika*, 2018.
[10] R. Zhong and G. Yue, “DDoS Detection System Based on Data Mining,” *Proc. Second Int. Symp. Netw. Netw. Secur. (ISNNS ’10)*, vol. 1, pp. 62–65, 2010.