Analisis Log Web Server dengan Pendekatan Algoritme *K-Means Clustering* dan *Feature Importance*

Asyrafi Adnil Ma'ali[1)], Girinoto[2)], Muhammad Novrizal Ghiffari[3)], Raden Budiarto Hadiprakoso[4)]

1) Rekayasa Keamanan Siber, Politeknik Siber dan Sandi Negara, asyraffi.adnil@student.poltekssn.ac.id

2) Rekayasa Perangkat Lunak Kriptografi, Politeknik Siber dan Sandi Negara, girinoto@poltekssn.ac.id

3) Rekayasa Perangkat Lunak Kriptografi, Politeknik Siber dan Sandi Negara, m.novrizal@student.poltekssn.ac.id

4) Rekayasa Perangkat Lunak Kriptografi, Politeknik Siber dan Sandi Negara, raden.budiarto@poltekssn.ac.id

Abstrak

Analisis log sering kali dibutuhkan pada kegiatan forensik setelah terjadi insiden serangan pada jaringan. Pada penelitian ini dilakukan analisis log untuk mencari anomali pada web server melalui pendekatan *unsupervised machine learning* dengan menggunakan algoritme *k-means clustering* yang diintegrasikan dengan *Elbow Method*. Sebelum dilakukan proses pembentukan klaster *data log* ditransformasi dalam serangkaian proses *feature extraction*. Untuk pemahaman lebih lanjut, pemanfaatan metode analisis *feature importance* digunakan untuk mengetahui *feature* mana yang paling dominan berperan penting dalam proses pembentukan klaster. Hasil *clustering* memberikan visualisasi terdapatnya klaster yang bersifat anomali dari klsster lainnya dan *feature* yang berperan penting dalam proses pembentukan klaster tersebut adalah *character_bigram*.

Kata kunci: *analisis log*, *clustering*, *Elbow Method*, *feature importance*, *k-means*.

Abstract

Log analysis is often required in forensic activities after a network attack incident has occurred. In this study, log analysis was carried out to look for anomalies on the web server through an unsupervised machine learning approach using the k-means clustering algorithm which was integrated with the Elbow Method. Prior to the cluster formation process, the log data is transformed in a series of feature extraction processes. For further understanding, the use of feature importance analysis method is used to determine which features have the most dominant role in the cluster formation process. The results of clustering provide a visualization of the presence of anomalous clusters from other clusters and the feature that plays an important role in the process of forming the cluster is character_bigram.

Keywords: log analysis, clustering, Elbow Method, feature importance, k-means.

1. PENDAHULUAN

Umumnya suatu sistem *Intrusion Detection System* (IDS) dipasang untuk mendeteksi intrusi yang masuk ke dalam jaringan atau *website* yang dikelola, namun tidak menutup kemungkinan terdapat kejadian salah prediksi *False Positive* (FP) ataupun *False Negative* (FN). Cara untuk memastikan adanya anomali yang terjadi perlu dilakukan pemeriksaan manual data log web server yang mencatat setiap transaksi di web server atau proses analisis log. Menurut NIST SP 800-92 [1], Administrator Jaringan dan Sistem suatu organisasi belum tentu memiliki kemampuan untuk melakukan analisis log secara mendetail karena banyaknya data yang ada pada log web server [1].

Log analysis tools banyak ditemukan baik yang bersifat komersial maupun *opensource* salah satunya adalah Elasticsearch, Logstash, and Kibana Stack (ELK Stack). ELK Stack melakukan proses analisis log terbatas untuk mengetahui ringkasan statistik data *raw data* dari sebuah log yang masih berbentuk teks (non-numerik). Proses untuk mengetahui apa yang terjadi di dalam *raw data log* tidak ditampilkan, sehingga masih menyulitkan bagi administrator untuk melakukan analisis.

Salah satu pendekatan dalam analisis log pada dekade terakhir ini adalah dengan menggunakan, *machine learning* [2]. Salah satu metode dalam *unsupervised-machine learning* yang bisa digunakan adalah metode *clustering*, yaitu metode untuk membagi satu set data menjadi beberapa kelompok yang memiliki kemiripan karakteristik antar satu dan lainnya. Salah satu dari metode *clustering* yang sederhana dan popular adalah Algoritme *k-means*. Uddhav Raj et al. [3] dan Zulfadilah et al. [4] memanfaatkannya dalam upaya menganalisis log web sever dengan melakukan perbandingan dari beberapa algoritme *clustering*, akan penentuan inisiasi banyak *k-cluster* masih bersifat sembarang dan tidak ada analisis lebih lanjut untuk menyelidiki hasil *feature extraction* yang berperan penting dalam penentuan hasilnya.

Berdasarkan uraian di atas, penelitian ini mencoba menyempurnakan hasil kerja Uddhav Raj et al. [3] dan Zulfadilah et al. [4], melalui *improved k-means* dengan cara mengintegrasikan dengan Elbow Method merujuk model Syakur *et al* [5] dan Nainggolan *et al* [6]. Sebagai analisis lanjutan akan diterapkan *Feature Importance Analysis* untuk mengetahui *feature* mana yang berperan penting sebagai informasi tambahan dalam pembentukan

cluster.

2. LANDASAN TEORI

Pada bagian ini akan disampaikan beberapa landasan teori yang menjadi acuan dalam penelitian ini, mulai dari analisis log web server, Elbow Method, *k-mean clustering* dan analisis *feature importance*.

2.1. Analisis *Log Web Server*

File log web server merupakan sebuah file teks biasa sederhana yang mencatat informasi transaksi setiap kali pengguna meminta *request* pada situs web [7]. File ini dibuka saat layanan web server dimulai dan tetap terbuka saat server merespons permintaan pengguna. Sumber utama dari data mentah (*raw data*) adalah web server log yang selanjutnya akan disebut sebagai file log [8]. Bagi administrator web, file log sangat membantu untuk mengetahui informasi seperti: halaman mana dari website yang diminta; jenis *error* apa yang dialami pengguna; status yang dikembalikan oleh server dari *request* pengguna; dan berapa banyak *byte* yang dikirim dari server ke pengguna.

Menganalisis data yang ada memberikan banyak informasi tentang apa yang terjadi pada web server. Format dari file log juga memiliki beberapa jenis. Format yang paling umum digunakan adalah Format IIS, Format *W3C extended* dan NCSA umum dan combined. Untuk file log yang digunakan dalam penelitian ini menggunakan format IIS dengan rincian format yaitu *IP address* dari pengguna, *Request date and time*, URL dan *Status code* HTTP. Seperti yang ditunjukkan pada Gambar 1.

```
IP,Time,URL,Staus
10.128.2.1,[29/Nov/2017:06:58:55,GET /login.php HTTP/1.1,200
10.128.2.1,[29/Nov/2017:06:59:02,POST /process.php HTTP/1.1,302
10.128.2.1,[29/Nov/2017:06:59:03,GET /home.php HTTP/1.1,200
10.131.2.1,[29/Nov/2017:06:59:04,GET /js/vendor/moment.min.js HTTP/1.1,200
10.130.2.1,[29/Nov/2017:06:59:06,GET /bootstrap-3.3.7/js/bootstrap.js HTTP/1.1,200
10.130.2.1,[29/Nov/2017:06:59:19,GET /profile.php?user=bala HTTP/1.1,200
10.128.2.1,[29/Nov/2017:06:59:19,GET /js/jquery.min.js HTTP/1.1,200
10.131.2.1,[29/Nov/2017:06:59:19,GET /js/chart.min.js HTTP/1.1,200
10.131.2.1,[29/Nov/2017:06:59:30,GET /edit.php?name=bala HTTP/1.1,200
10.131.2.1,[29/Nov/2017:06:59:37,GET /logout.php HTTP/1.1,302
10.131.2.1,[29/Nov/2017:06:59:37,GET /login.php HTTP/1.1,200
10.130.2.1,[29/Nov/2017:07:00:19,GET /login.php HTTP/1.1,200
10.130.2.1,[29/Nov/2017:07:00:21,GET /login.php HTTP/1.1,200
10.130.2.1,[29/Nov/2017:13:31:27,GET / HTTP/1.1,302
10.130.2.1,[29/Nov/2017:13:31:28,GET /login.php HTTP/1.1,200
10.129.2.1,[29/Nov/2017:13:38:03,POST /process.php HTTP/1.1,302
10.131.0.1,[29/Nov/2017:13:38:04,GET /home.php HTTP/1.1,200
10.131.0.1,[29/Nov/2017:13:38:07,GET /contestproblem.php?name=RUET%20OJ%20Server%20Testin
10.130.2.1,[29/Nov/2017:13:38:19,GET / HTTP/1.1,302
```

Gambar 1. *Raw Data Log Web Server*

2.2. Feature Extraction

Feature Extraction merupakan sebuah proses dekomposisi untuk membuat sebuah data menjadi beberapa variabel kolom atau *feature* dalam matriks yang nantinya akan diproses oleh model pengolahan [9]. Hal tersebut dapat berupa satu atau beberapa jenis *feature*. Satu jenis *feature* dapat mencakup beberapa atribut. Dalam kebanyakan kasus, satu jenis *feature* yang diekstraksi akan digabungkan menjadi sebuah matriks. Dalam matriks ini, setiap baris mewakili satu transaksi file log, dan setiap kolom mewakili satu atribut dari *feature* [9].

Karena log dapat dilihat sebagai data tekstual, beberapa teknik pemrosesan dengan menggunakan pendekatan ranah *Natural Language Processing* (NLP) dapat diterapkan. Secara umum pendekatan mengekstrak teks kedalam bentuk frekuensi *N-gram* dari log. Dimana *N-gram* adalah urutan *n* item yang berdekatan dari urutan teks yang diberikan [10].

Berikut merupakan hasil *feature extraction* yang akan digunakan dalam penelitian ini yang digunakan juga dalam penelitian [3]:

a. *Character-bigrams*, merupakan jumlah karakter pada suatu array yang diambil tiap dua karakter.
b. *Character-ngrams*, merupakan jumlah karakter pada suatu array yang diambil tiap N karakter.
c. *Character-trigrams*, merupakan jumlah karakter pada suatu array yang diambil tiap tiga karakter.
d. *Count_of_most_visited_page*, penghitungan dari banyaknya halaman yang dikunjungi.
e. *Number_of_records*, jumlah dari data dalam suatu interval yang diterapkan.
f. *Status*, merupakan status yang ditampilkan.
g. *Time_Difference_Maximum*, selisih waktu maksimal dari suatu IP mengakses sebuah link.
h. Time_Difference_Mean, selisih waktu rata-rata dari suatu IP mengakses sebuah link.
i. *Time_Difference_Sum*, jumlah waktu dari suatu IP mengakses sebuah link.
j. *Time_Difference_Variance*, variasi waktu maksimal dari suatu IP mengakses sebuah link.
k. *IP_rep*, reputasi dari alamat IP yang dicatat dalam log server.
l. *Word-count*, jumlah kata yang muncul dalam satu interval waktu.

2.3. Algoritme *K-Means Clustering*

Algoritme *k-means* termasuk dalam *unsupervised-machine learning* dan *distance based-clustering* yang outputnya membagi data observasi ke dalam *k-cluster* dan memerlukan penentuan awal atau inisiasi nilai *k centroid* sembarang. Gabungan *Elbow Method* dan *k-means* ditujukan untuk dapat menentukan nilai *k* atau banyak *cluster* yang ideal. Berikut adalah tahapan algoritme yang digunakan [5], [6]:

1. Cari *k* sebagai kandidat banyak *cluster* yang akan dibentuk terbentuk. Penelitian ini akan menggunakan *Elbow Methods Criterion* untuk memilih jumlah *k-cluster* yang akan digunakan untuk pengelompokan data pada algoritme *k-means*. *Elbow Methods Criterion* dinyatakan dengan *Sum of Squared Error* (SSE).

$$SSE = \sum_{k=1}^{K} \sum_{x_i \in K} \|x_i - C_k\|_2^2$$

   dengan k = banyaknya *cluster* terbentuk, *Ci* = *cluster* ke - *i*, *x* = data yang muncul setiap *cluster*.

2. Tentukan titik pusat (*centroid*) *cluster* di awal secara acak. Penentuan *centroid* awal dilakukan secara random dari objek yang tersedia sebanyak *k cluster*, kemudian untuk menghitung *centroid i-cluster* berikutnya, dengan rumus sebagai berikut:

$$v = \frac{\sum_{i=1}^{n} x_i}{n} \qquad i=1,2,..,n$$

3. Hitung jarak setiap objek ke setiap *centroid* menggunakan *Euclidean distance*.

$$d(x,y) = \|x - y\|$$

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2 \,; i = 1,2,3,\ldots,n}$$

dengan x_i = variabel pada objek x ke-i dan y_i = *output* y, n = banyak objek.

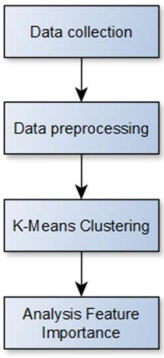
4. Alokasikan setiap objek ke *centroid* terdekat.
5. Alokasi objek ke dalam setiap *cluster* pada iterasi dengan algoritme *k-means*. Ukur jarak kedekatan setiap objek anggota *cluster* dengan titik pusat *cluster*.
6. Lakukan iterasi, kemudian proses penentuan posisi *centroid* baru dengan menggunakan persamaan pada poin kedua.
7. Ulangi langkah 3 jika posisi *centroid* baru dengan *centroid* lama tidak sama.

2.4. *Feature Importance*

Analisis *feature importance* umum digunakan dalam pemodelan *supervised-machine learning* untuk mengevaluasi *feature* model klasifikasi berupa nilai *scoring* dari bobot suatu *feature* yang memberikan dampak dalam model [11]. Terdapat beberapa pendekatan dalam memperoleh *feature importance* skor, salah satunya adalah dengan pendekatan *Mean Decrease in Impurity* (MDI) dari Random Forrest [12], [13].

3. METODOLOGI

Metodologi yang digunakan pada penelitian ini menggunakan proses dalam penggunaan *machine learning* secara umum seperti pada Gambar 2.



Gambar 2. Tahapan Penelitian

Pada tahap *data collection*, sebagaimana tujuan dari penelitian ini akan menyempurnakan penelitian Uddhav Raj et al. [3] dan Zulfadilah et al. [4], maka *dataset* dan beberapa bagian *data preprocessing* yang berisi transformasi *raw data* dan *feature extraction* merujuk pada penelitian [3] dengan menambah beberapa bagian tertentu seperti *word-count* dan *bigram-word count*. Pada tahap *preprocessing* ini juga, penerapan *normalization* dilakukan untuk menghilangkan satuan unit dari *feature* yang dihasilkan. Di tahap implementasi algoritme *k-means clustering,* diawali dengan penentuan banyak *k-cluster* yang ideal yang disarankan dari hasil *Elbow method*. Ditahap terakhir setelah *output* hasil *clustering k-means* diperoleh, akan digunakan sebagai label/target model *supervised-machine learning* untuk memperoleh skor *feature Importance.*

Dari hasil *cluster* dan *feature importance* yang diperoleh nantinya akan digunakan sebagai bahan pemeriksaan manual (*expert judgment*) atas fenomena anomali yang ditemukan. Analisis dilakukan dengan mengkombinasikan pengetahuan dari jenis-jenis serangan yang ada.

4. HASIL DAN PEMBAHASAN

Data log web server yang masih dalam bentuk file *extensi* *.txt diubah ke dalam bentuk data frame. Data yang berbentuk numerik seperti alamat *IP*, *Time* dan *Status* dilakukan penyesuaian penulisan seperti alamat *IP* akan dihilangkan titiknya, *time* akan diambil tanggal dan waktu Jam:Menit:Detik.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15788 entries, 0 to 15787
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   IP      15788 non-null  object
 1   Time    15788 non-null  object
 2   URL     15788 non-null  object
 3   Staus   15788 non-null  int64
dtypes: int64(1), object(3)
memory usage: 493.5+ KB
```

Gambar 3. *Data Properties* Hasil *Feature Extraction Process*

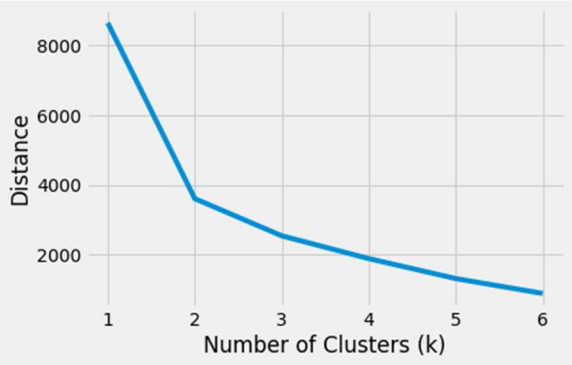
Gambar 3 menunjukan bahwa dari data log web server yang diperoleh menunjukan berjumlah 15.788 *row* transaksi. Selanjutnya disederhanakan berdasarkan agregasi transaksi per-interval satuan waktu jam. Berikut metadata dari hasil proses *feature extraction* berdasarkan agregasi per-jam jejak transaksi yang di sajikan pada Gambar 4. Pada proses tersebut menghasilkan tabel baru dengan dimensi 665 *row* dari 13 *feature*.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 665 entries, 0 to 664
Data columns (total 13 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Character-bigrams             665 non-null    int64
 1   Character-ngrams              665 non-null    int64
 2   Character-trigrams            665 non-null    int64
 3   Word-count                    665 non-null    int64
 4   Count_of_most_visited_page    665 non-null    int64
 5   Number_of_records             665 non-null    int64
 6   Status                        665 non-null    int64
 7   Time_Difference_Maximum       665 non-null    int64
 8   Time_Difference_Mean          665 non-null    float64
 9   Time_Difference_Sum           665 non-null    int64
 10  Time_Difference_Variance      665 non-null    float64
 11  Character-bigrams-Word-Count  665 non-null    int64
 12  IP_rep                        665 non-null    float64
dtypes: float64(3), int64(10)
memory usage: 67.7 KB
```

Gambar 4. *Data Properties* Hasil *Feature Extraction Process*

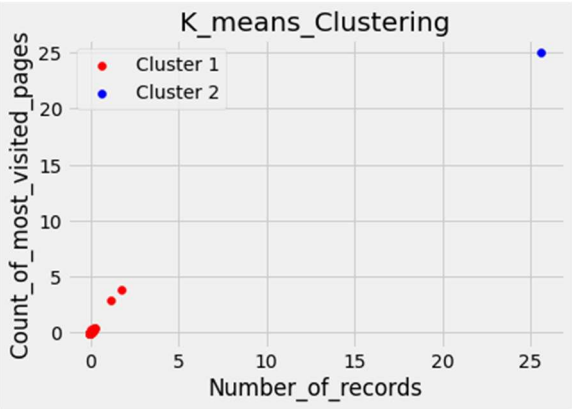
4.1. Hasil *Elbow Methods Criterion* dan *k-means*

Hasil *Elbow Methods Criterion* untuk simulasi *k-means* dari beberapa kandidat nilai *k* disajikan dalam grafik *distance* atau SSE yang disajikan pada Gambar 5. Dari Gambar 5 dapat dilihat bahwa *Elbow Methods Criterion* memberikan indikasi nilai *k* ideal jatuh pada nilai $k = 2$.



Gambar 5. Grafik Hasil *Elbow Method*

Setelah diperoleh nilai *k* yang disarankan pada proses sebelumnya, maka dilakukan proses *clustering k-means* dengan menggunakan input $k = 2$. Hasil *cluster* yang diperoleh ditunjukan pada Gambar 6.



Gambar 6. Hasil Clustering *k*=2

Dari proses *clustering* didapatkan hasil bahwa *cluster* 2 hanya berisi 1 buah *row* yaitu pada observasi ke-562. Hal ini sangat berbeda dengan apa yang terjadi pada *cluster* 1 yang berisi 664 *row*.

	Character-bigrams	Character-ngrams	Character-trigrams	Word-count	Count_of_most_visited_page	Number_of_records	Status
562	1185	284	579	35995	1841	5092	200

Time_Difference_Maximum	Time_Difference_Mean	Time_Difference_Sum	Time_Difference_Variance	Character-bigrams-Word-Count	IP_rep
86399	176.022	896128	9.194734e+06	37180	1012821.0

Gambar 7. Observasi yang dinyatakan sebagai *Cluster* 2 oleh *k-means*

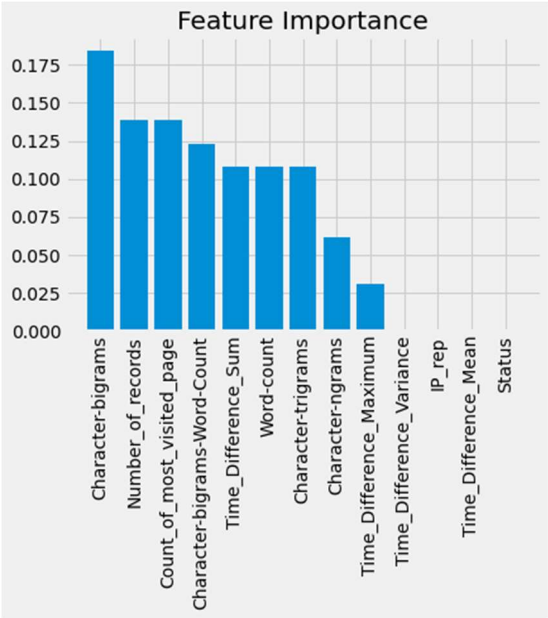
Gambar 7 menggambarkan karakteristik hasil *clustering* pada tabel agregasi transaksi di interval waktu tersebut. Jika dilihat dengan seksama pada Gambar 8, sampel *raw data* log menunjukan adanya transaksi anomali waktu akses dari *IP* 10.128.2.1, 10.130.2.1 dan 10.131.0.1 secara berturut-turut. Dari data tersebut, dapat disimpulkan adanya usaha masuk secara terus menerus yang dilakukan oleh beberapa *IP* atau bisa disebut sebagai *Distributed Denial-of-Service* (DDoS).

4.2. *Feature Importance*

Gambar 9 menunjukkan hasil dari *feature importance* dengan nilai tertinggi 0.18402 didapatkan oleh *character-bigrams* yang disusul oleh *number_of_records*. Dari hasil tersebut pula diketahui bahwa *feature*: *status*, *time_different_mean*, *time_different_variace* serta *IP_rep* berturut-turut memiliki kontribusi terendah. Atas dasar uraian tersebut, dapat diambil catatan bahwa pada proses *feature extraction* selanjutnya dapat disarankan tidak perlu mengikutsertakan *feature*: *status*, *time_different_mean*, *time_different_variance* serta *IP_rep* karena tidak cukup memberikan variasi informasi yang dibutuhkan dalam proses *clustering*.

9634	10.128.2.1	[29/Jan/2018:20:20:25	GET /home.php HTTP/1.1	302
9635	10.130.2.1	[29/Jan/2018:20:20:25	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9636	10.128.2.1	[29/Jan/2018:20:20:25	GET /login.php HTTP/1.1	200
9637	10.128.2.1	[29/Jan/2018:20:20:25	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9638	10.128.2.1	[29/Jan/2018:20:20:25	GET /home.php HTTP/1.1	302
9639	10.130.2.1	[29/Jan/2018:20:20:25	GET /home.php HTTP/1.1	302
9640	10.131.0.1	[29/Jan/2018:20:20:25	GET /home.php HTTP/1.1	302
9641	10.131.0.1	[29/Jan/2018:20:20:25	GET /home.php HTTP/1.1	302
9642	10.131.0.1	[29/Jan/2018:20:20:26	GET /login.php HTTP/1.1	200
9643	10.131.0.1	[29/Jan/2018:20:20:26	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9644	10.128.2.1	[29/Jan/2018:20:20:26	GET /login.php HTTP/1.1	200
9645	10.131.0.1	[29/Jan/2018:20:20:26	GET /login.php HTTP/1.1	200
9646	10.131.0.1	[29/Jan/2018:20:20:26	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9647	10.128.2.1	[29/Jan/2018:20:20:26	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9648	10.131.0.1	[29/Jan/2018:20:20:26	GET /home.php HTTP/1.1	302
9649	10.131.0.1	[29/Jan/2018:20:20:26	GET /login.php HTTP/1.1	200
9650	10.131.0.1	[29/Jan/2018:20:20:26	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9651	10.131.0.1	[29/Jan/2018:20:20:28	GET /home.php HTTP/1.1	302
9652	10.131.0.1	[29/Jan/2018:20:20:28	GET /login.php HTTP/1.1	200
9653	10.128.2.1	[29/Jan/2018:20:20:29	GET /home.php HTTP/1.1	302
9654	10.131.0.1	[29/Jan/2018:20:20:29	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9655	10.128.2.1	[29/Jan/2018:20:20:29	GET /login.php HTTP/1.1	200
9656	10.128.2.1	[29/Jan/2018:20:20:29	GET /home.php HTTP/1.1	302
9657	10.128.2.1	[29/Jan/2018:20:20:29	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9658	10.128.2.1	[29/Jan/2018:20:20:29	GET /home.php HTTP/1.1	302
9659	10.128.2.1	[29/Jan/2018:20:20:29	GET /login.php HTTP/1.1	200
9660	10.128.2.1	[29/Jan/2018:20:20:29	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9661	10.128.2.1	[29/Jan/2018:20:20:29	GET /login.php HTTP/1.1	200
9662	10.128.2.1	[29/Jan/2018:20:20:29	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9663	10.131.0.1	[29/Jan/2018:20:20:30	GET /home.php HTTP/1.1	302
9664	10.131.0.1	[29/Jan/2018:20:20:30	GET /login.php HTTP/1.1	200
9665	10.128.2.1	[29/Jan/2018:20:20:30	GET /home.php HTTP/1.1	302
9666	10.131.0.1	[29/Jan/2018:20:20:30	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9667	10.130.2.1	[29/Jan/2018:20:20:30	GET /home.php HTTP/1.1	302
9668	10.130.2.1	[29/Jan/2018:20:20:30	GET /login.php HTTP/1.1	200
9669	10.130.2.1	[29/Jan/2018:20:20:30	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9670	10.131.0.1	[29/Jan/2018:20:20:30	GET /home.php HTTP/1.1	302
9671	10.131.0.1	[29/Jan/2018:20:20:30	GET /home.php HTTP/1.1	302
9672	10.131.0.1	[29/Jan/2018:20:20:30	GET /login.php HTTP/1.1	200
9673	10.131.0.1	[29/Jan/2018:20:20:31	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9674	10.131.0.1	[29/Jan/2018:20:20:31	GET /home.php HTTP/1.1	302
9675	10.131.0.1	[29/Jan/2018:20:20:31	GET /login.php HTTP/1.1	200
9676	10.131.0.1	[29/Jan/2018:20:20:31	GET /login.php HTTP/1.1	200
9677	10.131.0.1	[29/Jan/2018:20:20:31	GET /home.php HTTP/1.1	302
9678	10.130.2.1	[29/Jan/2018:20:20:31	GET /home.php HTTP/1.1	302
9679	10.130.2.1	[29/Jan/2018:20:20:31	GET /home.php HTTP/1.1	302
9680	10.130.2.1	[29/Jan/2018:20:20:32	GET /login.php HTTP/1.1	200
9681	10.128.2.1	[29/Jan/2018:20:20:32	GET /home.php HTTP/1.1	302
9682	10.128.2.1	[29/Jan/2018:20:20:32	GET /login.php HTTP/1.1	200
9683	10.128.2.1	[29/Jan/2018:20:20:32	GET /js/vendor/modernizr-2.8.3.min.js HTTP/1.1	200
9684	10.128.2.1	[29/Jan/2018:20:20:32	GET /home.php HTTP/1.1	302

Gambar 8. Sampel Transaksi pada Interval Waktu pada *raw log web server*



Gambar 9. *Feature Importance*

5. KESIMPULAN

Penelitian ini berusaha menunjukan proses analisis *log web server* melalui pendekatan metode *clustering* untuk menangkap fenomena anomali pada *raw data* log dengan menggunakan algoritme *k-means* yang dikombinasikan oleh *Elbow Method* dan pengetahuan (*knowledge expertise*) dari karakteristik dan jenis-jenis serangan pada infrastruktur jaringan komputer. Selanjutnya, melalui hasil analisis *feature importance* dapat ditunjukkan bahwa tidak semua kandidat *feature* dalam proses *feature extraction* dapat memberikan cukup kontribusi informasi dalam proses pembentukan *cluster*.

REFERENSI

[1] National Institute of Standards and Technology, NIST SP 800-92: Guide to Computer Security Log Management, Gaithersburg: U.S. Department of Commerce, 2006.

[2] Q. Cao and Y. Qiao, "Machine Learning to Detect Anomalies in Web Log Analysis", 3rd IEEE International Conference on Computer and Communications, pp. 519-523, 2017.

[3] U. Raj, A. Kumar, M. R. Ajit and T. Ashutosh, "Log analysis using distributed system using MapReduce and Hadoop", National Institute of Technology Calicut, pp. 1-7, 2018.

[4] Zulfadhilah, M., Prayudi, Y., & Riadi, I. Cyber Profiling Using Log Analysis And K-Means Clustering. International Journal of Advanced Computer Science and Applications, 2016.

[5] Syakur, M.A., Khotimah, Rochman & Satoto, B.D., "Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster", IOP Conference Series: Materials Science and Engineering, 2018

[6] Nainggolan, R., Perangin-angin, R., Simarmata and Tarigan, A.F., "Improved the Performance of the K-Means Cluster Using the Sum of Squared Error (SSE) optimized by using the Elbow Method", Journal of Physics: Conference Series, 2019.

[7] T. A. Cahyanto and Y. Prayudi, "Investigasi Forensika Pada Log Web Server untuk Menemukan Bukti Digital Terkait dengan Serangan Menggunakan Metode Hidden Markov Models," Seminar Nasional Aplikasi Teknologi Informasi, pp. 15-19, 2014.

[8] K. R. Suneetha and D. R. Krishnamoorthi, "Identifying User Behavior by Analyzing Web Server Access Log File," International Journal of Computer Science and Network Security, vol. IX, no. 4, pp. 327-332, 2009

[9] Ghojogh, B., Samad, M. N., Mashhadi, S. A., Kapoor, T., Ali, W., Karray, F., & Crowley, M., "Feature selection and feature extraction in pattern analysis: A literature review", arXiv preprint :1905.02845, 2019.

[10] J. Habdak, N-gram based Text Categorization, Bratislava: Comenius University Faculty of Mathematics, Physics And Informatics Institute Of Informatics, 2005.

[11] Saarela, M., Jauhiainen, S., "Comparison of feature importance measures as explanations for classification models.", SN Appl. Sci. 3, 272, https://doi.org/10.1007/s42452-021-04148-9, 2021.

[12] Breiman, L.,"Random Forests." Machine Learning 45, 5–32, https://doi.org/10.1023/A:1010933404324, 2001.

[13] Scikit-learn documentation, "Permutation Importance vs Random Forest Feature Importance (MDI)", https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance.html.