Clustering Analysis for Malware Behavior Detection using Registry Data

Nur Adibah Rosli[1], Warusia Yassin[2], Faizal M.A[3], Siti Rahayu Selamat[4]
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Malaysia

***Abstract*—The increase of malware attacks may increase risk in information technology industry such as Industrial Revolution 4.0 that consists of multiple sectors especially in cyber security. Because of that malware detection technique plays vital role in detecting malware attack that can give high impact towards the cyber world. In accordance with the technique, one of unsupervised machine learning able to detect malware attack by identifying the behavior of the malware; which called clustering technique. Owing to this matter, current research shows a paucity of analysis in detecting malware behavior and limited source that can be used in identifying malware attacks. Thus, this paper introduce clustering detection model by using K-Means clustering approach to detect malware behavior of data registry based on the features of the malware. Clustering techniques that use unsupervised algorithm in machine learning plays an important role in grouping similar malware characteristics by studying the behavior of the malware. Throughout the experiment, malware features were selected and extracted from computer registry data and eventually used in the proposed clustering detection model to be clustered as normal or suspicious behavior. The results of the experiment indicates that this proposed model is capable to cluster normal and suspicious data into two separate groups with high detection rate which is more than 90 percent accuracy. Ultimately, the main contribution based on the findings is the proposed framework can be used to cluster the data with the use of data registry to detect malware.**

***Keywords*—Malware; malware detection; behavior analysis; k-means clustering; data registry**

I. Introduction

Developing a malware detection system model by using K-Means clustering approaches is demanding in IDS field. Even though clustering techniques causes a number of advantages in grouping similar malware characteristics, however unsupervised algorithm specifically against registry information is absent in machine learning techniques. First, clustering is one of the best method in recognizing similar binaries and put them in one group as used by [1]–[4]. Other researchers [4]–[9] shows that recognizing the malware in malware analysis by using K-Means clustering method is the best way. However, none of them use this method using registry information to analyze the malware. Thus, based on that matter, there is still low significant of malware analysis in this field. Second, Malware analysis by using registry information has been explored by previous researchers [10]–[15] with different methods of malware detection. Yet, K-Means algorithm is still not an alternative to cluster malware data to detect any malware causes low significant malware detection method. Even so, the use of K-Means clustering as malware detection in windows registry has been review by [16] in their survey and K-Means clustering method seems promising in malware detection field. Thus, this paper addresses the two issues, which are lack of data in detecting malware behavior and lack of further analysis in detecting malware behavior. K-Means clustering detection model with appoint of data mining, peculiarly clustering method is a notable field that can be explored to overcome this matter. It is a need to have continuous of IDS improvement in term of the accuracy of malware analysis, the detection time and the suitable detection approach; are the motivations for this research. Therefore, the objective of this research is to generate registry information in detecting malware behavior and secondly to propose clustering analysis against registry information for malware detection. This research focuses on the K-Means clustering as a method to analyze malware in windows registry, which accurately identify normal and suspicious behavior with minimum false positive and false negative as well as maximum true positive and true negative. In addition, the detection method is designed such that it could operate accurately in identifying intrusion in host-based intrusion detection system (HIDS).

Universiti Teknikal Malaysia Melaka (UTeM) and Cyber Security Malaysia (CSM) under CMERP Grant –GLUAR/CSM/2016/FTMK-CACT/100013.

II. Literature Review

A. Malware

Over the years, many security problems escalated because of excessive use of Internet usage and computer systems over network. Briefly, the interconnected systems such as Web servers, database servers and cloud computing servers are exposed to many threads that come from cyber attackers. Regarding this concern, CERT statistics [17] shows that the amount of intrusion every year, and distressingly, they keep growth excessively. Consequently, the attacks in the form of malicious intrusion exposing the network to vulnerabilities that causes serious impact to computer and information system besides violated the policies of computer security such as CIA or Confidentiality, Integrity and Availability. Malicious intrusion known as Malware is an intrusive software, which can be in term of file or code causing harm. It is a program that has malicious intentions, which is created and designed for clear objective to get access in information system without permission from the administrator [18]. This refers to [19] that described malware as developed malicious software which has an intention of lunching malignant tasks. Similar to [20], defined that malware is a type of program by accomplishing

something as such the attacker needs it to be. It also defined decades ago by [21] supported by [22] as a malicious software that fulfill harmful intention of an attacker such as "Viruses", "Worms" and "Trojan horses". In addition, there are many types of malware nowadays that has been created to take advantage as well as harming others such as "Botnet", "Adware", "Spyware" and "Ransomware". Different category of malware act differently compare to each other:

1) Virus: A virus can be defined as a program that can 'infect' other programs by modifying them to comprise a possibly evolved version of itself [23]. Author in [24] also described virus as malicious program software that can replicate themselves and spread among computers. To be simplified, virus means a simple form of software that is loaded and launched without user's permission while reproducing itself or infecting other software.

2) Worm: Similar to virus, worm can spread over the network but the different is it can replicate to other software. It is a self-replicating computer program and uses a network to send copies of itself to other nodes such as computers on the network without any user involvement [25]. Author in [26] also defined worms are generally self-propagating software, since they self-propagate but usually rely on the receiving user to activate them.

3) Trojan: Different to virus and worm, Trojan is a type of malware that appear as legitimate software. This malware class is used to define the malware types that aim to appear as genuine software. Because of this, the general spreading vector utilized in this class is social engineering, i.e. making people think that they are downloading the legitimate software [27].

4) Botnet: Botnet is an infected network of computers on the Internet with software robots, which is called as bots [28]. [29] Described botnets are large collections of computers called "zombies" that are under the control of a single attacker. Botnet also defined by [30] as collection of computer that has been infected by malicious software a; converts bots, drones, or zombies, which have been integrated into a bigger collection through a centralized command and control infrastructure. It means that in an infected computer, the information systems build a network of bots that receive instructions from a server known as command-and-control server.

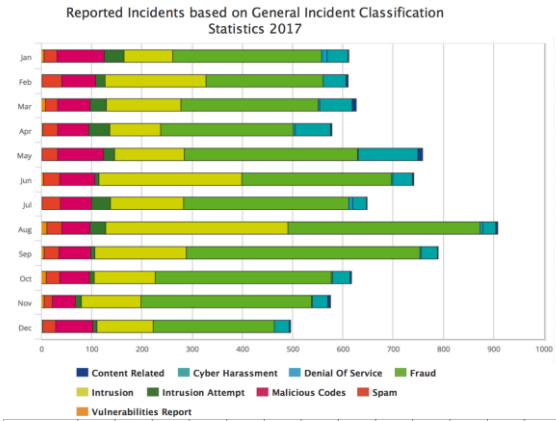
5) Adware: Adware is an advertising supported programming that performs its activity by displaying or downloading the advertisement to a user computer after the installation of malicious application or programming [31]. Author in [32] also described that an ad-injecting browser extension such as of adware, analyzing all the malware activities of ad injecting extension also falls under the category of adware. The main purpose of adware is displaying advertisements on the computer and can lead to dramatic results. Adware are basically an applications which has a goal in getting maximal revenue to the developer while giving the user the minimal amount of value [33].

6) Spyware: A spyware is a malware; which follow the action of user silently without the client knowing. Standard actions of spyware are tracking search history to send personalized advertisements as well as tracking activities and afterwards selling them to the third parties. The gathered information can include the website, browser and system information which are visited by user. Spyware can likewise control over the framework [31].

7) Ransomware: Ransomare is a type of malware that aims to encrypt all the data on the machine and ask a victim to transfer some money as the ransom to get the decryption key. Usually, a machine infected by ransomware is "frozen" as the user cannot open any file, and the desktop picture is used to provide information on attacker's demands. The Netskope Cloud Report of September 2016 revealed that 55.9% of malware-infected files found in cloud apps are shared publicly thus, the cloud is an attractive platform for attackers [34]. Currently, ransomware is a major threat faced by organizations and individuals alike. Ransomware is part of a recent malware trend that prevents or limits access to resources in the infected computer [35], Ranomware can be detected in registry as discussed by [36].

Data from the Malaysia Computer Emergency Response Team (MyCERT) shows the reported incidents of cyber-attacks [17]. Malware attacks rely under malicious codes and it is crucial incidents as it is in the top three of the statistics. Total cyber-attacks incidents in year 2017 are presented in Fig. 1.

Besides, statistic form Information Security Timelines and Statistics [37], in January 2018 malware hit a new maximum rate with 43.5%, which is the highest rate compared to the other attack vectors. Fig. 2 shows the attack vector statistic in January 2018.



	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	TOTAL
Content Related	2	5	9	2	9	2	1	4	2	2	5	3	46
Cyber Harassment	41	45	64	71	119	39	27	25	32	36	31	30	560
Denial of Service	11	0	3	3	1	3	8	6	2	2	1	0	40
Fraud	296	233	274	265	346	298	329	382	466	351	340	241	3821
Intrusion	98	201	148	101	138	284	146	363	181	121	119	111	2011
Intrusion Attempt	39	19	32	41	22	8	37	31	8	9	11	9	266
Malicious Code	94	68	65	62	92	71	62	66	64	60	46	74	814
Spam	26	38	24	30	31	32	36	30	29	26	17	25	344
Vulnerabilities Report	5	2	8	3	1	4	2	11	6	10	5	3	60
TOTAL	612	611	627	578	759	741	648	908	790	617	575	496	7962

Fig. 1. Statistic of Reported Incident, 2017.

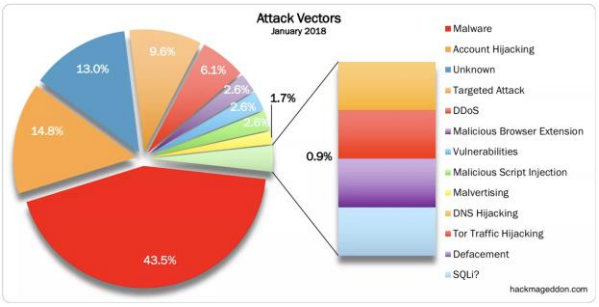


Fig. 2. Attacks Vector Statistic, January 2018.

Cyber-attacks become the biggest threat in computer and networks system around the world. Because of that it is important to merge IDS that can detect and analyze the data with high accuracy (i.e., true positives and negative) and low false detection (i.e., false positive and negative) in the minimal detection time. So, K-Means clustering detection model with appoint of data mining, peculiarly clustering method is a notable field that can be explored to overcome this matter. It is a need to have continuous of IDS improvement in term of the accuracy of malware analysis, the detection time and the suitable detection approach; are the motivations for this research.

B. Malware Detection

Malware interrupt the file registry when entering a computer and basically malware tend to create and modify computer files system and Windows registry entries besides the computer interprocess communication and basic network interaction [21]. Intrusion attack such as malwares are known to breach the policy of network security in organizations and continuously tries to interrupt the core fundamental of cybersecurity which are Confidential, Integrity and Availability or known as CIA. Therefore, previous cybersecurity researcher has proposed detection-based for malware intrusion, which is a framework that monitors the behavior of system activity. Then, the behavior will be analyzed by the framework and notify the users if there is a sign of intrusion. Furthermore, [24] define intrusion detection as an event monitoring process where it can be implemented in network or computer system and capable to send notification if there is any intrusion has been detected. Besides, scanning is vulnerability evaluation that has been used in intrusion detection to access the vulnerabilities over computers system or network. There are two types of attack that can be analyze by intrusion detection such an intrusion attack can be explained as an attack mainly from malware infect machine or network outside organization while on contrary, misuse attack can be define as an attack targeting within organization. In a nutshell, intrusion detection capable to monitor system activities including scanning vulnerabilities, system integrity and configuration, recognizing each attack patterns, analyzing irregular activities in operating system and tracing users if the users have break or violate policies.

At any cost, the detection of malware is important and crucial as it conquer more than half of malware attack that exploit on the computer registry; and it can be detected by using Intrusion Detection System as the early defense over the malware attack [36]. One of the solutions in detecting any intrusions is an Intrusion Detection System (IDS) to avoid the network and computer system from any cyber attack. There are many function and tasks in system security as discussed in most studies related to detection system. The uses of IDS protect computer system and ultimately improve the system security.

An IDS is a real time system that can monitor in spite of analyze network packet and system audit log for detecting and identifying malware behavior or any intrusion attempts in a computer system, before sending intrusion alerts to system administrators [38]. Another definition by [39] described that. IDS is a specific hardware and software that can monitor any occurring event automatically in computer system or network that collect and synchronize the event records as well as analyze them if there is any sign of security violation. Thus, an IDS is a type of security hardware and software that designed to give alert to the administrators automatically if there is any attacks, security policy violations or malicious activities that can compromise information system by monitoring system activity through examining vulnerabilities in the system and analyzing the vulnerabilities patterns bases on the mechanism of detection of previous attacks. In addition, IDS able to monitor the Internet search automatically to find any latest threats that could be future attacks based on the malware behavior.

In term of malware detection, this paper focusing on anomaly detection as it detects any intrusion through analyzing and make divergence from the pattern of the normal behavior [38]. Anomaly is an abnormality to a normal behavior and profiles pattern, which is a show as normal or usual behaviors. The anomaly detection is a derivation of information from monitoring regular computer activities, network connections, hosts or users over a specific time and sometimes it is called as Behavior-based Detection [40]. The profile of malware can be either static or dynamic, but still creating many attributes. After that, anomaly detection differentiates the normal profiles with the observed events to recognize significant attacks. As stated by [41] based on the abnormality anomaly-based intrusion detection characterize the baseline models that are normal and identifies any attacks from the models. This method is able to identify any unique attacks and capable to target a wide range of attacks. As an example, by is using low-level architectural and microarchitectural features that available from HPCs which is hardware performance counter, Tang also examine that the feasibility and limits of performing anomaly-based malware detection. Furthermore, based on the principle of detection, anomaly-based detection can be more promising techniques in discovering any computer intrusion or attacks [42] in term of monitoring and flags any network activities displaying significant deviation from legitimate traffic profiles as suspicious objects. Thus, this research used the concept of anomaly detection based on the benefit of anomaly detection to detect previous unknown intrusion or attack.

C. Malware Detection in Windows Registry

Windows Registry is known as hierarchical of database information, setting, option, and other value of hardware and software that stored in low-level settings of Microsoft Windows operating system. It can be accessed through registry key that analogous to file system directories [13]. Windows and every program are continually referencing the registry;

hence when there are any changes of them, changes are also made to suitable areas in the registry.

Detection of malware in windows registry by using K-Means clustering is a new topic in this field. Even though the previous researchers had discovered malware in the same location, which is in registry, however they have used different detection method as malware detection apart from K-Means clustering method.

In AccessMiner, which is using system-centric models achieve a large-scale collection for malware protection examine the diversity of system calls by [10]. The analysis of the data presents that simple malware detector by using alternative detection model that characterizes the general interactions between benign programs and the operating system (OS). The system-centric approach models analyze benign programs that access OS resources such as files and registry entries and it results in raising very few or even zero false positives malware detection.

Similar with Behavior-based Detection Model, HOLMES [11] also analyze files and registry by using another model. It presents an automatic technique to extract optimally discriminative specifications, which uniquely determine a class of the programs. The proposed technique is based on graph mining and concept analysis, scales to large classes of programs due to probabilistic sampling of the specification space. The proposed HOLMES can synthesize discriminative specifications that accurately distinguish between programs, sustaining an 86% detection rate on new, unknown malware, with 0 false positives rate.

Other than that, by using behavioral sequential patterns as malware detection method, [12] proposed dynamic malware detection system based on mining the API sequences and iterative patterns extracted from an executable trace of API calls. The framework is able to examine and detect malicious behavior as well as introduced the concept of iterative pattern mining in this field.

Moreover, to detect malware in virtual environment based on its behavior, a dynamic malware analyzer has been proposed [13]. This approach is able to bypass anti-VM detection technique in detecting malware and their behavior and also identify the technique that has been used by malware. The dynamic malware analyzer can monitor the resources of the system for example, network connection, file system, processes and also services. Then, it gives information regarding the malware attack to analyst and noted the changes of Windows registry. Finally, the accuracy test by using Pahadus public malware set is successful with high detection ratio which is 92%.

In experimental analysis of ransomware by [14], when a computer machine is attacked by ransomware, the analysis of ransomware, basically focus on the families evolution and characteristic of the. Ransomware interaction with the file system, registry activities, and network operations. The experimental results show that the detection of ransomware is achievable based on examining abnormal file system and registry activities in Windows environment. To check the effectiveness of the experiment, a computer machine is already had all inbuilt security procedures upgraded and running, which automatically detect and delete all those ransomware variants before it is infected by ransomware.

The last method in malware analysis and detection tools discussed by [15]. The author did an analysis of by comparing the well-known malware and benign programs. In the experiment, samples are taken which is 100 malware and 100 benign programs that come from many different sources and have been analyzed by using different type of Windows machines versions. The test results indicate that it is extremely difficult to detect the presence of malware by using only one tool. Thus, the new approach is by using both dynamic and static analysis tools; it can increase the detection rate as well as its accuracy.

The comparison of related works under detection malware on windows registry presented in Table I shows that different detection models are used in the same malware location, which is windows registry by different authors.

The comparison result shows that K-Means clustering is a new approach in detecting malware on windows registry. In addition, significant malware analysis by using K-Means clustering in high demand in analyzing malware accurately as well as better malware detection. All previous researchers had explored about K-Means clustering and applied the algorithm in different model to achieve the objectives. K-Means clustering is widely used in many different areas in detecting malware.

TABLE. I. PREVIOUS WORK (DETECTION OF MALWARE IN WINDOWS REGISTRY)

Author	Detection Model	Detection Location	Input Data	Result
[10]	System-Centric Models	Windows: system call, registry	Data collection (system call)	Increase false positive rate
[11]	Behavior-based Detection Model, HOLMES	Windows: registry, system call	Real malware samples (Honeypot)	Low false positive rate
[12]	Behavioral Sequential Patterns	Windows: file system, process, registry	Logging calls (API call)	Low false positive rate
[13]	Dynamic malware Analyzer	Windows: connections, processes, registry, file operations	Pahadus public malware set sample	High true positive rate
[14]	Analyzing samples of selected ransomware variants	Windows: file system, registry, network activity	Malware data sets collection (virus total, public malware repositories, security forums)	High true positive rate
[15]	Static and dynamic analysis tools	Windows: file system, registry, process activity	Program sample malware (Windows)	High true positive rate

D. Malware Detection by using K-Means Clustering

K-Means clustering is a method of cluster analysis in which the defined 'k' is separating the clusters with the existence of center value between all the grouped objects. However, in data mining perspective, the implemented K-Means clustering algorithm separates the time interval between the normal and abnormal data in the same training dataset. Differ from database manners, clustering can be referred as the capability of many servers or instances to connect to one database while in IDS, clustering technique is usually use within anomaly detection in exploring group of malware data information without knowing the former relationship knowledge of the data. So, clustering method clusters the objects according to their characteristic of data points, in such every single data point in a cluster is identical to those in the same cluster, but diverse from another clusters [43]. For this reason, clustering is one of the most admired concepts in the domain of unsupervised learning as the anomaly detection is generally unsupervised detection. The idea is the same data points tend to belong to same groups or clusters, as identified by the distance of the data from the local centroids. Fig. 3 shows the example of clustering in a graph.

The graph shows that there are only two centroids, which are marks as 'X'. The 'X' mark depends on the number of cluster that is defined in the first step of the process. The resulting cluster centroids are then used for fast anomaly detection in monitoring of new incoming data [44]. The K-Means clustering algorithm is one of the simplest unsupervised learning algorithms as shown in Fig. 4 that resolves the clustering problem [8] by:

1) Collecting dataset of malware.
2) Identifying the number of clusters (*k*).
3) Initializing the k centroids (k-means) for the data.
4) Determining the distance of each malware from each centroid and then assign each malware to the cluster with centroid closest to it.
5) Recounting the centroids for each cluster.
6) Steps 4 and 5 are repeated until there is no change in cluster centroids.
7) If formed clusters do not look reasonable, repeat the steps 1-6 for different number of clusters.

In Fig. 4, clustering of data begins with identifying the number of cluster according to the characteristics of the data. Then the centroid of each cluster will be decided and accordingly, the distance of each data will be determined starting from the selected centroid. The data, which is in the minimum distance with the centroid, can be considered as the designed group otherwise it will be excluded from the group. The used of malware data in clustering method is suitable as malicious data characteristics almost similar to normal data, however somehow it cannot hide its different behavior compared to normal data behavior. Thus, the different behavior can be detected by using this clustering method. In data science, K-Means clustering is a type of algorithm that has been used to utilize the method of vector quantization specifically in signal processing field and most of the time examiners use it to solve clustering problems. According to [18], in K-Means clustering method, the whole dataset are transform to Voronoi cells by taking observations and finally create the '*k*' groups in which every observation is a segment of a computed nearest mean cluster. It means that it creates 'k' similar clusters of data points and the data instances that fall outside of these groups could potentially be marked as anomalies. Thus, K-Means is a widely used clustering algorithm and this algorithm can be said as the most popular clustering algorithm among the geometric procedures [45] because of its computational simplicity, efficiency and ease of implementation [43], [46]. As it is straightforward algorithm, the computational time is faster then the other algorithm, thus the time of malware clustering process can be minimized [47].

Thus, K-Means Clustering detection method has been the focus of this research based on the motivation. In improving malware behavior detection, clustering analysis is a need by means of K-Means clustering as a new detection method especially in detecting malware.

The uses of DNS as carrier for its command and control determines and reverse engineered Feederbot, which is a botnet [5]. K-Means clustering is combined with Euclidean Distance based classifier correctly classified more than 14m DNS transactions of 42,143 malware samples concerning DNS-C&C usage then, uncovers another bot family with DNS C&C. In addition, this method correctly detected DNSC&C in mixed office workstation network traffic. For instance, DNS C&C provide a mechanism to detect DNS C&C in network traffic.

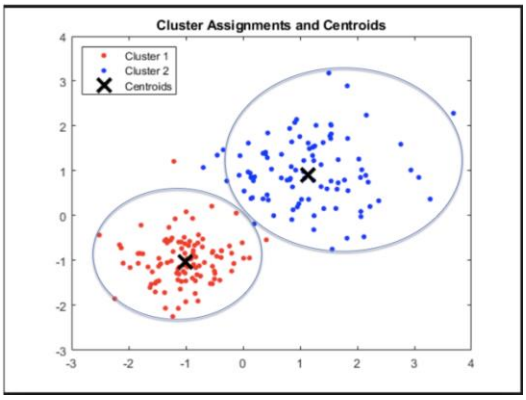


Fig. 3. Graph of Cluster Assignment and Centroid.

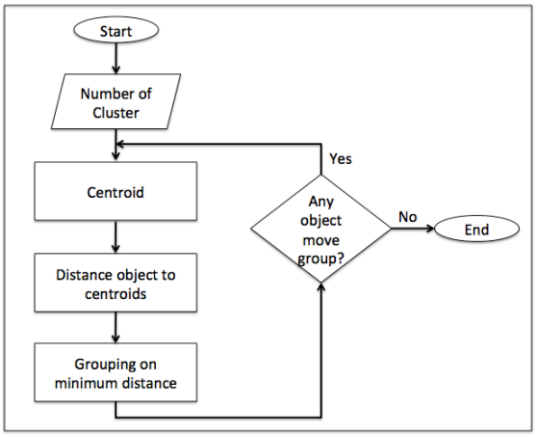


Fig. 4. Clustering Algorithm Flow Process.

In addition, a large-scale log analysis known as Beehive can detect suspicious activities in enterprise network has been proposed by [6]. This approach uses K-Means algorithm to solve data mining problem by extracting the information from data long produced by large enterprise. The improvised of signature-based technique can accurately detect many attack incidents and identify any suspicious behavior if data information. On top of that, Beehive evaluates the collected data log from the large enterprise and compares all the incidents to identify if there is any malicious events and violation of policies without being detected.

Apart from Beehive, the proposed semi-supervised approach only by using K-Means approach can self-merge the information of unknown malware which is unlabeled data into detection system as discussed by [7]. The semi-supervised approach extracts the information of the cluster before inserts the information into the SVM, which is support vector machine classification system by applying global K-Means clustering algorithm. As result, the experiment shows that the proposed approach reach high accuracy rate of detection compared to the existing supervised approach.

In genetic boosting classification to detect malware, the use a static analysis approach has been proposed by [4] can removes the samples that cannot be classified with adequate firmness and need. Thus, with the used of K-means clustering algorithm, the sample can be grouped into regions according to the features. Next, genetic algorithm guided the boosting process, to execute in every region besides evaluated using a test dataset discarding those regions, which do not reach a minimum accuracy threshold.

Malware detection using genetic algorithm (GA) optimized K-means and Hidden Markov Model (HMM) [8] considering the interrelated problem of malware classification. The HMMs is instructed for a variety of malware generators and variety of compilers. As per results, further classification was done using K-Means algorithm with GA in HMM. The GA tuned k-means clustering, and this approach is suggested for better malware detection.

Meanwhile, Distributed GHSOM which is Growing Hierarchical Self Organizing Maps is an unsupervised clustering algorithm for Big Data has been proposed by [9]. To fulfill the requirement on tolerance of variation between samples, the proposed method clusters the data samples dynamically. It pretends as an attractive unsupervised learning solution for data, which have finite information to decide the number of clusters in advance. It used parallel computation with scala actor models for GHSOM construction, distributing vertical and horizontal expansion tasks to actors and showing significant performance improvement.

In short, by using K-Means clustering approach in malware detection model may increase the accuracy of malware detection. However the used of this previous methods not truly focuses on specific malware location. Thus, this research will focus on K-Means clustering at Windows registry.

The comparison of related works under K-Means clustering algorithm as malware analysis can detect malware accurately is presented in Table II. The entire method listed in the table shows the result in term of detection rate which are true positive, true negative, false positive and false negative. As expected, the result is promising of the use of K-Means clustering in detecting malware accurately as proved by [47].

TABLE. II. PREVIOUS WORK (MALWARE DETECTION BY USING K-MEANS CLUSTERING)

Author	Detection Model	Method Used	Input Data	Result
[5]	DNS for Command and Control	K-Means clustering and Euclidean Distance based classifier	Malware binaries	High true positive rate.
[6]	Beehive: Large-Scale Log Analysis	K-Means clustering	Data collections (Log)	High true positive rate.
[4]	Genetic Boosting Classification	K-means clustering algorithm	Binary files	Low false positive rate.
[7]	Semi-supervise approach by using global K-Means clustering	K-Means clustering	Executable files (Win32-based systems)	High true positive rate.
[8]	GA optimized K-means and Hidden Markov Model (HMM)	Genetic Algorithm (GA) K-mean clustering	Pre-obtained dataset of malwares (API and opcode)	Low false positive rate.
[9]	Growing hierarchical self organizing maps (GHSOM)	K-means clustering algorithm	Malware executable (OWL database and Windows APIs)	Low false positive rates.

III. METHODOLOGY

The methodology of this paper consists of three steps started with data collection, followed by data preparation, and lastly data analysis. In data collection, the malware binary files are downloaded from the trusted website and sorted before they are used for the experiment. After the selection, the data was prepared running selected malwares in control environment and the registry features are extracted. Lastly, the extracted features are combined in a database for data analysis. All the process implemented based on the proposed Clustering Detection Model.

Fig. 5 shows the process flow of data clustering method that proposed in this study. There are four phases; which are binary execution phase, file extraction phase, registry data extraction phase and clustering phase. It is started with process 1 which is extracting normal file of registry from virtual machine. The process continued with process 2 by downloading binary file and injects it into the same virtual machine. Then, process 3 is extracting the infected file of registry from the virtual machine. In process 4, all the files are stored in a database and in process 5; which is registry data extraction phase, registry data is extracted and prepared. After that, process 6 begins by clustering all the data files by using K-Means clustering in clustering phase. Lastly, in process 7, the output data will be updated in the different table in the same database. All the processes are classified into four main

phases and the phases of the detection model describes as follows:

Phases 1: Binary Execution Phase

In this phase, the binary file is run in virtual machine that is Drakvuf environment. Then, all the activities are captured as log format.

Phases 2: File Extraction Phase

Then, all the data, which is the malware activities are extracted in this phase. There are two types of data that are extracted; first, default file (normal activities) and second infected file (suspicious activities).

Phases 3: Registry Data Extraction Phase

After that, all the collected registry data is extracted and prepared in this phase, as the extracted data are imbalance data.

Phases 4: Clustering Phase

The last phase is clustering phase in which the balanced data is analyzed by using K-Means clustering algorithm to cluster the data either it is malware or not. Euclidean Distance formula is used to measure the distance of centroid and data points. The formula is shown in Fig. 6.

A. Binary Execution Phase

This phase is started by downloading binary files from the trusted website. The binary files of the malware are downloaded from trusted website which is mlawares.com (https://www. malwares.com/). Apart from containing malware binary files, malwares.com able to analyze various advance, newborn, mutated malicious code and URLs. So, this website is suitable for malware analysis as it provides latest malware for this project. The malwares are downloaded from this website for this experiment. Then the malwares are checked in the Lastline portal to ensure that the malware can be run in Windows 7 that suits for the experiment, which only focuses on Windows 7 operating system. The selected malwares and 2 normal files are uploaded into the database. One malwares contains around half million of data. After that, all the binaries are executed in Drakvuf environment.

B. File Extraction Phase

File extraction phase started by extracting default file, which is normal behavior, and extracting infected files, which is suspicious behavior from virtual machine that is Drakvuf environment. Drakvuf is a virtualization based agentless black-box binary analysis system that allows in-depth execution tracing of arbitrary binaries, which include operating system.

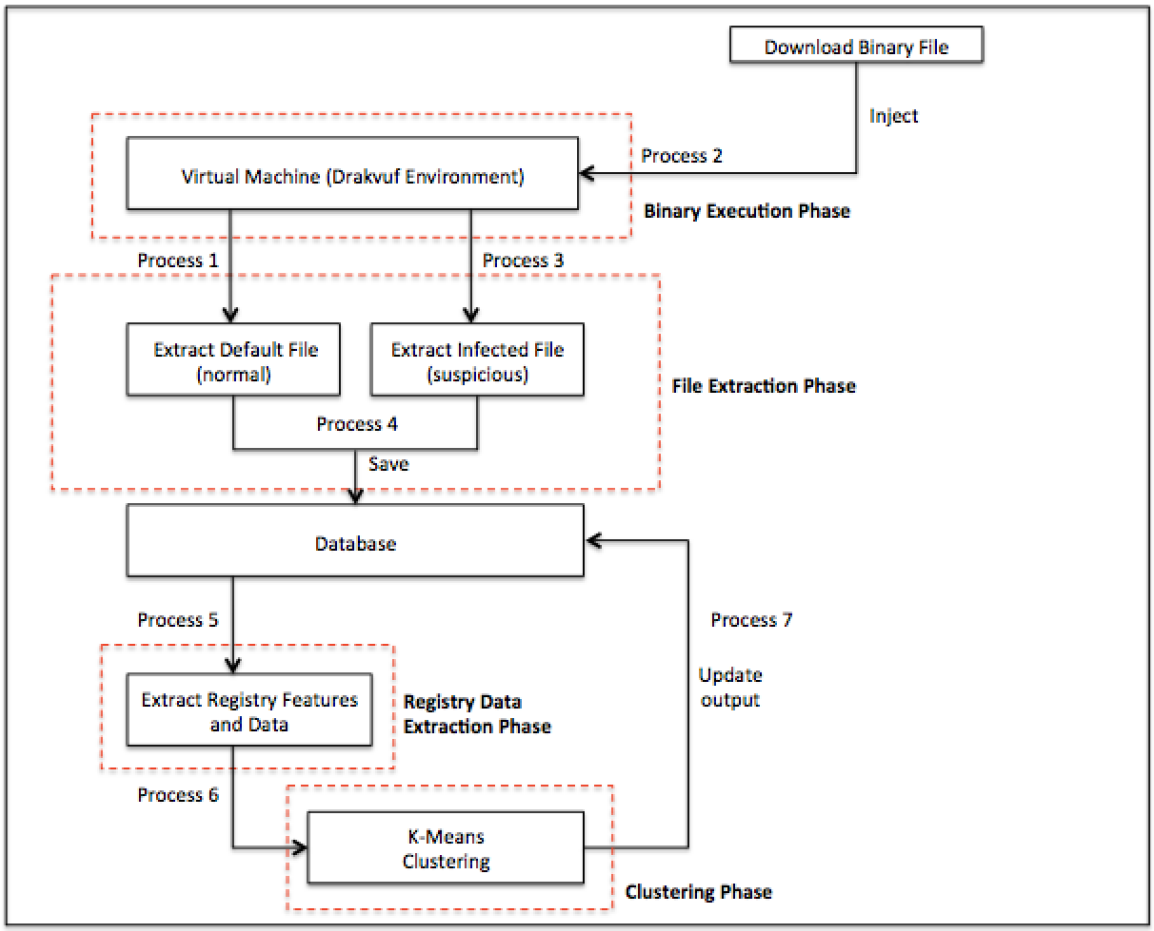


Fig. 5. Clustering Detection Model.

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

Fig. 6. Euclidean Distance Formula.

C. Registry Data Extraction Phase

This phase started with uploading the data into database. By using SQL query, the data is selected based on targeted data for data analysis. The RegUtil system is used to upload the srp file of normal and infected file. It connected to database to store the data. The normal and infected files are uploaded through RegUtil system into a database. In the database, the data was extracted to get the only used data for the experiment by using SQLYog database. Then, the data is chunked into several paths before it can be used for analysis. To checked whether the path is correct or not, it was checked with regedit software.

D. Clustering Phase

Using Waikato Environment analyzes clustering phase started with uploading the prepared data for Knowledge Analysis (Weka) for clustering method implementation. The version of Weka used is 3.8.2. Weka is an information mining programming that uses an accumulation of machine learning calculations and the calculations can be associated direct to the information or called from the Java code. The apparatuses that can be utilized as a part of the information accumulation are relapse, grouping, affiliation, information pre-preparing, arrangement, and perception. In this undertaking, the instrument that had been utilized zone characterization. The balanced data is analyzed by using K-Means clustering algorithm to cluster the data either it is malware or not.

The performance of K-Means clustering detection in the field of Intrusion Detection is usually assessed using the following measurements:

- True Positive (TP) is the number of malware samples that has been detected accurately.
- True Negative (TN) is the number of normal samples that has been detected accurately.
- False Positive (FP) is the number or normal samples that is falsely detected as an attack.
- False Negative (FN) is the number of malware samples that is falsely detected as normal.

The detection rate is calculated by using the formula: Detection Rate = (TP) /(TP+FP)[1]. In addition, to detect intrusion attempt for unsupervised data, the features of the malware are needed, as there is no numeric number that can be used to calculate the distance between the points. Besides that, Elbow method is used to determine the value of K as well as the stop point after the result is plotted in the graph as discussed by [48]. To be exact, Elbow method is a method of interpreting and validating the consistency of the cluster analysis designed to help to find the best number of clusters in a dataset as shown in Fig. 7.

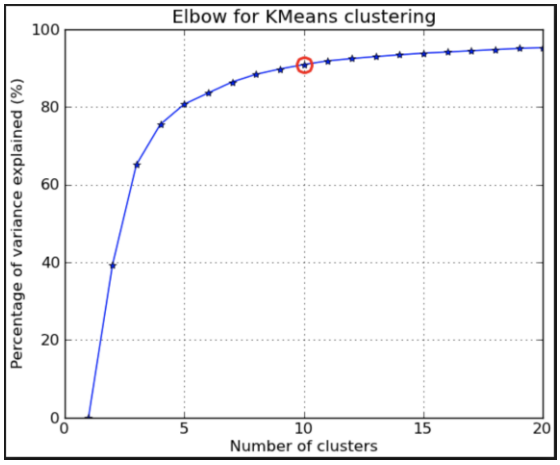


Fig. 7. Elbow Method Graph.

IV. Result, Analysis and Discussion

In this paper, the result of the experiment is based on the percentage of attack detection according to the number of cluster that has been identified during the experiment and it showed that cluster 20 has the highest percentage among the nine tested clusters, which is 96.41%. The percentage exceeds the acceptable detection rate, which is 90% for malware detection as stated by [18]. The percentage of attack is also presented in Fig. 8. It follows the principal of elbow method to find the value of K as stated in previous chapter. Thus, the number of K based on the result is 20, which is the best number that had the highest percentage of attack detection. In Table III, the high percentage of attack detection indicates that this method can highly detect the malware attack by clustering the normal and abnormal data. The normal data belongs to normal group while the abnormal data are excluded from the group. Thus the abnormal data are classified as malware as their behavior are different from the normal data and they are not in the same group with the normal group.

Based on the result, cluster 20 shows the best result among the other clusters. The detection rate is 96.41% however, the false alarm which is 3.55% causes the decreases number of malware detection. Because of that, some of the attacks can be missed to detect, as it is known as normal. Fig. 9 shows root cause of the miss-detected malware. Even though the label of the data is different for attack and normal, but all the features are same. Thus, it is hard to recognize the attack, as it is same with the normal features.

TABLE. III. Percentage of Attack Detection

Number of cluster	Percentage of Attack Detection (%)
5	76.48
10	80.78
15	92.9
20	96.41
25	92.23
30	92.43

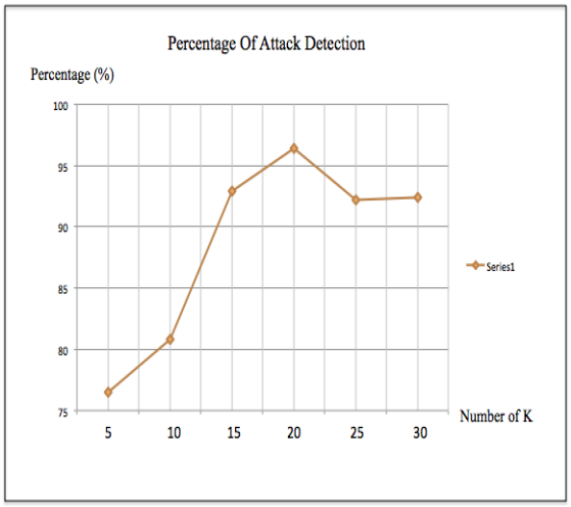


Fig. 8. Graph Percentage of Attack Detection.

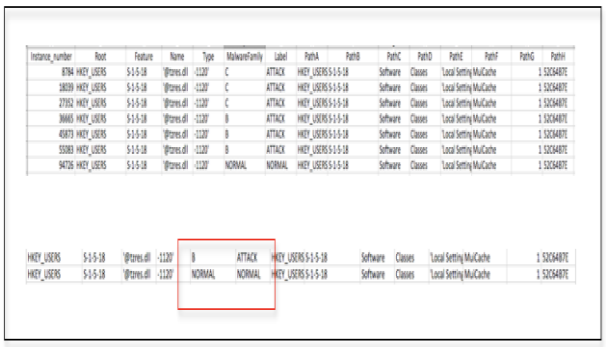


Fig. 9. Root Cause of the Miss-Detected Malware.

V. Conclusion and Future Works

Intrusion Detection System (IDS) is use as malware detector globally, causes many researchers in exploring this field. In this research project, clustering method is proposed for better malware detection. It is because of lack of analysis in detecting malware behavior causes low malware detection due to limited sources on this information especially in windows registry to identify malware activities. Clustering techniques that use unsupervised algorithm in machine learning plays an important role in grouping similar malware characteristics but this approach is absent in malware analysis environment specifically in registry information. Thus, the purpose of this research project is to study registry information and proposing clustering analysis against registry information to improve malware detection. Thus, the research project has been conducted successfully and proposing clustering analysis model against registry information to improve malware detection. Based on the result, the proposed method has a detection rate more than 90%. It is shows that the proposed method has high rate in detecting malware based on the features of the unknown file. According to the direction of this research project, it gives great benefit to community by providing guidance and steps on how to overcome the stated problem. Finally, it is a hope for community can fetch the importance of this research project and used it concerning to Information Technology and Computer Science area.

References

[1] J. Stiborek, T. Pevný, and M. Rehák, "Probabilistic analysis of dynamic malware traces," Comput. Secur., vol. 74, pp. 221–239, 2018.

[2] T. Wüchner, M. Ochoa, and A. Pretschner, "Malware detection with quantitative data flow graphs," pp. 271–282, 2014.

[3] A. D. James Baldwin, Omar Alhawi, "Leveraging Machine Learning Techniques for Windows Ransomware Network Traffic Detection," 2017.

[4] A. Martín, H. D. Menéndez, and D. Camacho, "Genetic boosting classification for malware detection," 2016 IEEE Congr. Evol. Comput. CEC 2016, pp. 1030–1037, 2016.

[5] C. J. Dietrich, C. Rossow, F. C. Freiling, H. Bos, M. Van Steen, and N. Pohlmann, "On botnets that use DNS for command and control," Proc. - 2011 7th Eur. Conf. Comput. Netw. Defense, EC2ND 2011, pp. 9–16, 2012.

[6] T. Yen, A. Oprea, and K. Onarlioglu, "Beehive: large-scale log analysis for detecting suspicious activity in enterprise networks," Proc. 29th Annu. Comput. Secur. Appl. Conf., pp. 199–208, 2013.

[7] S. Huda et al., "Defending unknown attacks on cyber-physical systems by semi-supervised approach and available unlabeled data," Inf. Sci. (Ny)., vol. 379, pp. 211–228, 2017.

[8] A. Chanana and S. Singh, "Malware Detection Using GA optimized K-means and HMM," pp. 355–362, 2017.

[9] C. H. Chiu, J. J. Chen, and F. Yu, "An Effective Distributed GHSOM Algorithm for Unsupervised Clustering on Big Data," Proc. - 2017 IEEE 6th Int. Congr. Big Data, BigData Congr. 2017, pp. 297–304, 2017.

[10] A. Lanzi, D. Balzarotti, C. Kruegel, M. Christodorescu, and E. Kirda, "AccessMiner: Using System-Centric Models for Malware Protection," ACM Conf. Comput. Commun. Secur. 2010, pp. 399–412, 2010.

[11] S. Jha, M. Fredrikson, M. Christodoresu, R. Sailer, and X. Yan, "Synthesizing near-optimal malware specifications from suspicious behaviors," Proc. 2013 8th Int. Conf. Malicious Unwanted Softw. "The Am. MALWARE 2013, pp. 41–50, 2013.

[12] M. Ahmadi, A. Sami, H. Rahimi, and B. Yadegari, "Malware detection by behavioural sequential patterns," Comput. Fraud Secur., vol. 2013, no. 8, pp. 11–19, 2013.

[13] A. Pektaş and A. Tankut, "A dynamic malware analyzer against virtual machine aware malicious software," Secur. Commun. NETWORKS, vol. 7, no. 12, pp. 2245–2257, 2014.

[14] Monika, P. Zavarsky, and D. Lindskog, "Experimental Analysis of Ransomware on Windows and Android Platforms: Evolution and Characterization," Procedia Comput. Sci., vol. 94, pp. 465–472, 2016.

[15] O. Aslan and R. Samet, "Investigation of Possibilities to Detect Malware Using Existing Tools," 2017 IEEE/ACS 14th Int. Conf. Comput. Syst. Appl., pp. 1277–1284, 2017.

[16] D. Ucci, L. Aniello, and R. Baldoni, "Survey on the Usage of Machine Learning Techniques for Malware Analysis," 2017.

[17] Anon, "Reported Incidents based on General Incident Classification Statistics 2016," Reported Incidents bas ed on General Incident C las s ification Statis tics 2 014, 2017. [Online]. Available: https://www.mycert.org.my/assets/graph/pdf/2014-1.pdf.

[18] K. Kosmidis and C. Kalloniatis, "Machine Learning and Images for Malware Detection and Classification," Proc. 21st Pan-Hellenic Conf. Informatics - PCI 2017, no. December, pp. 1–6, 2017.

[19] A. K. Ajay and J. C.D., "Automated multi-level malware detection system based on reconstructed semantic view of executables using machine learning techniques at VMM," Futur. Gener. Comput. Syst., vol. 79, pp. 431–446, 2018.

[20] A. Souri and R. Hosseini, "A state-of-the-art survey of malware detection approaches using data mining techniques," Human-centric Comput. Inf. Sci., vol. 8, no. 1, 2018.

[21] A. Moser, C. Kruegel, and E. Kirda, "Exploring multiple execution paths for malware analysis," Proc. - IEEE Symp. Secur. Priv., pp. 231–245, 2007.

[22] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic malware-analysis techniques and tools," ACM Comput. Surv., vol. 44, no. 2, pp. 1–42, 2012.

[23] J. Horton and J. Seberry, "Computer Viruses An Introduction," vol. 19, no. 1, pp. 122–131, 1997.

[24] L. X. Yang and X. Yang, "A new epidemic model of computer viruses," Commun. Nonlinear Sci. Numer. Simul., vol. 19, no. 6, pp. 1935–1944, 2014.

[25] B. Rajesh, Y. R. J. Reddy, and B. D. K. Reddy, "A Survey Paper on Malicious Computer Worms," Int. J. Adv. Res. Comput. Sci. Technol., vol. 3, no. 2, pp. 161–167, 2015.

[26] C. Smith, A. Matrawy, S. Chow, and B. Abdelaziz, "Computer Worms: Architectures, Evasion Strategies, and Detection Mechanisms," J. Inf. Assur. Secur., vol. 4, pp. 69–83, 2009.

[27] M. Moffie, W. Cheng, D. Kaeli, and Q. Zhao, "Hunting trojan horses," Proc. 1st Work. …, no. January 2006, pp. 12–17, 2006.

[28] W. Kim, O.-R. Jeong, C. Kim, and J. So, "On botnets," Proc. 12th Int. Conf. Inf. Integr. Web-based Appl. Serv. - iiWAS '10, no. 2, p. 5, 2010.

[29] R. Wash, "Incentive design for home computer security," CHI '07 Ext. Abstr. Hum. factors Comput. Syst. - CHI '07, p. 1681, 2007.

[30] R. Abdullah and M. Abdollah, "Revealing the Criterion on Botnet Detection Technique," IJCSI Int. J. Comput. Sci. Issues, vol. 10, no. 2, pp. 208–215, 2013.

[31] P. Jyotiyana and S. Maheshwari, "Intelligent Systems Technologies and Applications 2016," vol. 530, pp. 449–460, 2016.

[32] X. Xing et al., "Understanding Malvertising Through Ad-Injecting Browser Extensions," Proc. 24th Int. Conf. World Wide Web - WWW '15, pp. 1286–1295, 2015.

[33] I. Ideses and A. Neuberger, "Adware detection and privacy control in mobile devices," 2014 IEEE 28th Conv. Electr. Electron. Eng. Isr. IEEEI 2014, 2014.

[34] A. Cohen and N. Nissim, "Trusted detection of ransomware in a private cloud using machine learning methods leveraging meta-features from volatile memory," Expert Syst. Appl., vol. 102, pp. 158–178, 2018.

[35] P. B. Pathak, "A Dangerous Trend of Cybercrime: Ransomware Growing Challenge," Int. J. Adv. Res. Comput. Eng. Technol., vol. 5, no. 2, pp. 371–373, 2016.

[36] R. Brewer, "Ransomware attacks: detection, prevention and cure," Netw. Secur., vol. 2016, no. 9, pp. 5–9, 2016.

[37] P. Passeri, "January 2018 Cyber Attacks Statistics," Information Security Timelines and Statistics, 2018. [Online]. Available: https://www.hackmageddon.com/2018/02/22/january-2018-cyber-attacks-statistics/.

[38] U. Ravale, N. Marathe, and P. Padiya, "Feature selection based hybrid anomaly intrusion detection system using K Means and RBF kernel function," Procedia Comput. Sci., vol. 45, no. C, pp. 428–435, 2015.

[39] S. C. Satapathy, B. N. Biswal, S. K. Udgata, and J. K. Mandal, "Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014," Adv. Intell. Syst. Comput., vol. 327, pp. 405–411, 2014.

[40] H. J. Liao, C. H. Richard Lin, Y. C. Lin, and K. Y. Tung, "Intrusion detection system: A comprehensive review," J. Netw. Comput. Appl., vol. 36, no. 1, pp. 16–24, 2013.

[41] A. Tang, S. Sethumadhavan, and S. J. Stolfo, "Unsupervised anomaly-based malware detection using hardware features," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8688 LNCS, pp. 109–129, 2014.

[42] Z. Zhiyuan Tan, A. Jamdagni, X. Xiangjian He, P. Nanda, and R. P. Ren Ping Liu, "A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 2, pp. 447–456, 2014.

[43] A. B. S. Serapião, G. S. Corrêa, F. B. Gonçalves, and V. O. Carvalho, "Combining K-Means and K-Harmonic with Fish School Search Algorithm for data clustering task on graphics processing units," Appl. Soft Comput. J., vol. 41, pp. 290–304, 2016.

[44] S. Agrawal and J. Agrawal, "Survey on anomaly detection using data mining techniques," Procedia Comput. Sci., vol. 60, no. 1, pp. 708–713, 2015.

[45] C. Bouveyron and C. Brunet-Saumard, "Model-based clustering of high-dimensional data: A review," Comput. Stat. Data Anal., vol. 71, pp. 52–78, 2014.

[46] P. Louvieris, N. Clewley, and X. Liu, "Effects-based feature identification for network intrusion detection," Neurocomputing, vol. 121, pp. 265–273, 2013.

[47] S. S. J and S. Pandya, "An Overview of Partitioning Algorithms in Clustering Techniques," vol. 5, no. 6, pp. 1943–1946, 2016.

[48] P. Bholowalia and A. Kumar, "EBK-Means : A Clustering Technique based on Elbow Method and K-Means in WSN," Int. J. Comput. Appl., vol. 105, no. 9, pp. 17–24, 2014.