CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

This chapter discusses methodology of this research including the explanation of used tools, methods, analysis, collection of data set and evaluation technique used.

3.2 Research Framework

The methodology and outcome for each objective is shown in the table 3.1 below.

Table 3.1: Mapping Research objectives, methodology and Outcomes

No	Objectives	Methodology	Outcome
1-	To propose new classification for the worm cloud based on cloud worm features	1. Collect cloud worm dataset 2. Create control lab environment (VM ware). 3. Install dynamic analysis tools 4. Inject cloud worm dataset into virtual cloud server. 5. Analyse and monitor cloud worm behavior by dynamic analysis. 6. Classify cloud worms based on infection, activation, payload, operating algorithm and propagation. 7. Data cleaning and Transformation	A cloud worm classification
2-	To develop a cloud worm detection technique by integrating the enhanced genetic algorithm	Improve existing genetic algorithm through enhancing selection proportional of fitness, tree crossover, tree mutation and evolution controller.	A cloud worm detection technique integrated with enhanced genetic algorithm.

3-	To propose a cloud worm response technique based on threat level.	**1.** The threat level is being measured based on the impact confidentiality, integrity and availability. **2.** Threat level measurement based on the assigned rules, weight and severity level measured by the security metrics.	A cloud worm response technique based on threat level.
4-	To evaluate the proposed cloud worm detection technique.	**1.** Simulation of the proposed technique and embedding it in Weka. **2.** Calculate the accuracy, TP and FP rate for the proposed and existing techniques and compare their performences.	A comparison result between proposed technique with the existing technique based on the accuracy rate

This research is also conducted in three different phases; initial study phase, dataset collection, analysis and classification phase, KDD process and implementation phase. The research processes are illustrated in Figure 3.1.

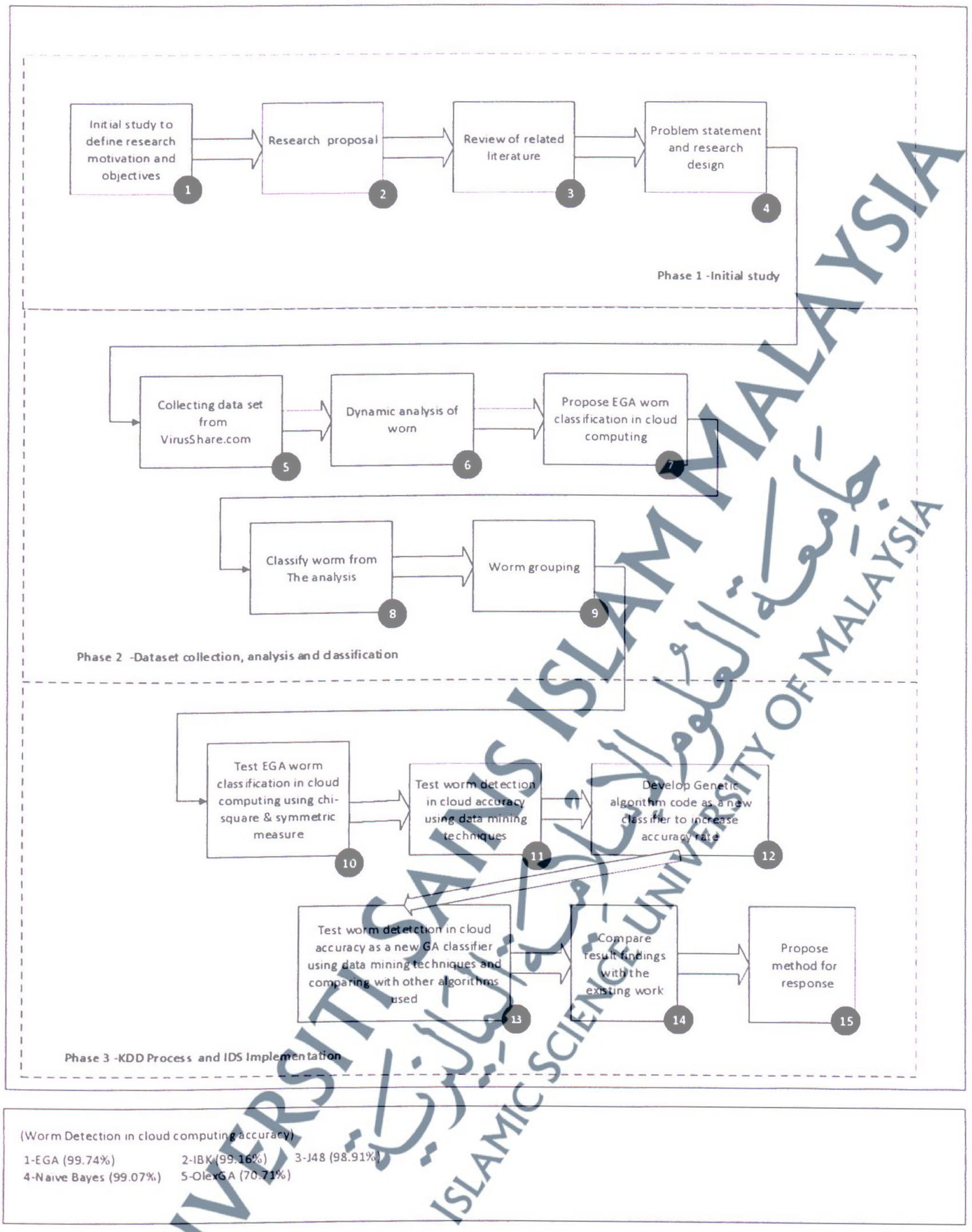


Figure 3.1: Overall research process for EGA technique.

3.3 Research Design

This section is to explain the research design and procedure of the research. Below are the steps that are involved in this research:

3.3.1 Worm Dataset

The dataset in this research consists of different types of worms and benign executables that are sourced from VirusShare (virusshare, 2015), Malwr website (malwr, 2016) and Arizona website (Arizona, 2015). This dataset is used to build up and evaluate the detection technique. The amount of worms' samples is 1018. From 23,699,719 samples available in virushare.com, these 1018 latest worm samples were collected which are identified by different anti-virus giant and Microsoft (virusshare, 2015). Virusshare is one of largest worm database freely available from the internet. According to virushare.com all collected samples are reported either in the year 2014 or 2015. Various researchers collected virus and worm samples for their research work from virusshare.com (Suarez-Tangil *et al.*, 2014; Jang *et al.*, 2015; Shijo & Salim, 2015; Kim & Kim, 2015).

On the other hand, the amounts of benign executables samples are 177. These benign executable samples had been used by many researchers including (Yadegari & Debray, 2015; Yadegari *et al.*, 2015; Lu & Debray, 2013).

Collected cloud worm samples are listed in Table 3.2 and visualised in Figure 3.2 with their types. Duplication of collected worm samples was avoided through MD5 key.

Table 3.2: List of collected cloud worm dataset

Worm Type	**No of worm (downloaded by type)**	**Total Downloaded**
Email-Worm	116	1018
Backdoor.Win32-Worm	56	
IM-Worm	42	
IRC-Worm	23	
Net-Worm	148	
P2P-Worm	106	
Worm.BAT	7	
I-Worm	57	
Win32.Worm.Downadup	39	
W32.ScriptDropperE.Worm	80	
Worm.Win32	344	

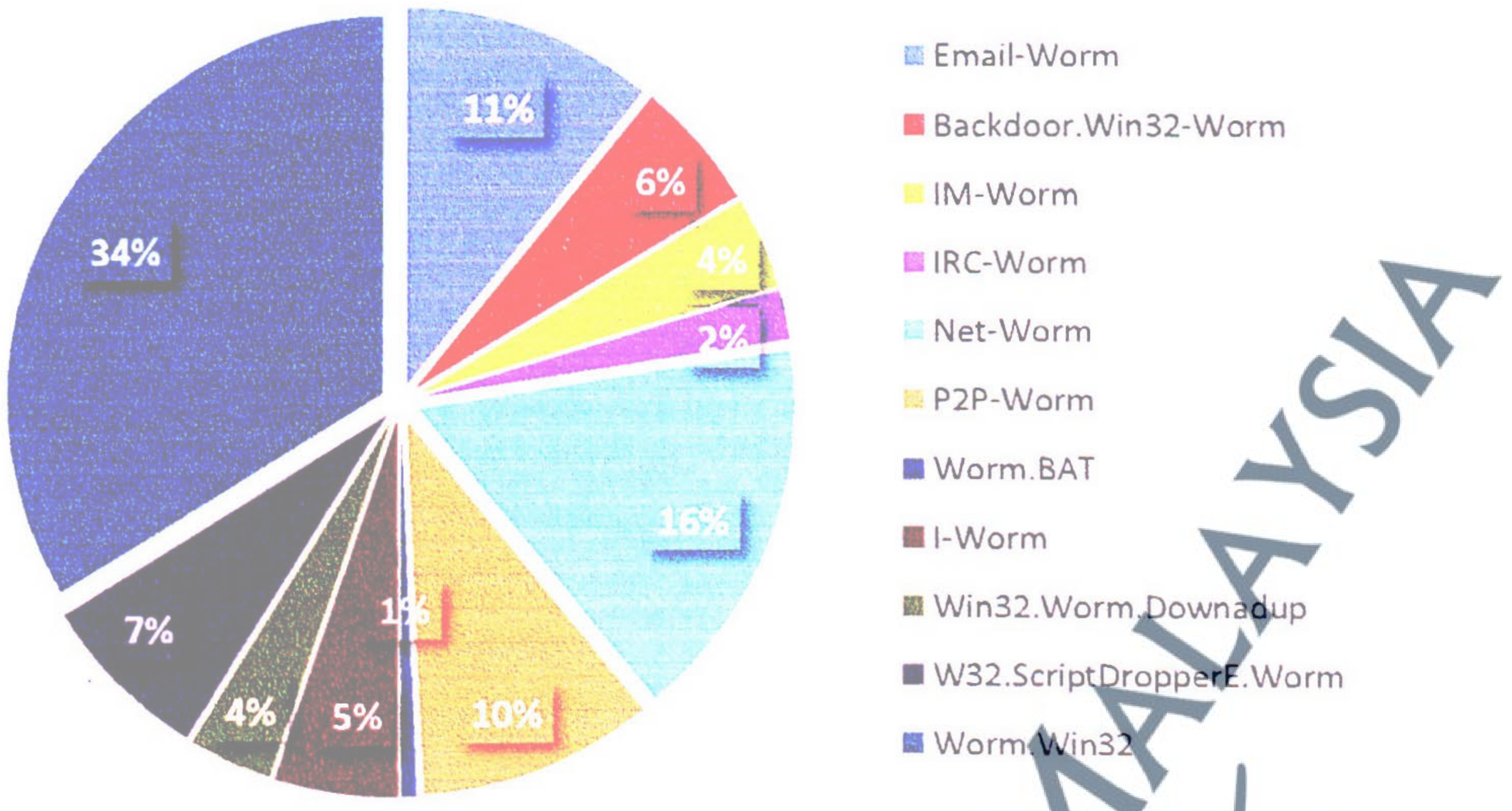


Figure3.2: Collected worm dataset by type.

3.3.2 Controlled Lab Architecture

The controlled architecture for laboratory environment is shown in Figure 3.3. To control worm propagation, it is necessary to disconnect the physical network connection in order to make a fully controlled isolated environment. But without network connection it is not possible to analyse worm samples dynamically. Due to this, virtualisation technology was used for implementing cloud environment virtually and other host (Attacker and monitoring host) connected to cloud through VMnet. Saudi *et al.*, (2009) used controlled lab architecture for worm analysis. Also, a research carried out by Abuzaid. (2013), also made use of controlled laboratory architecture to do their analysis on Trojan Horse malware. Likewise, controlled lab architecture was also used in this work. SANS (Sanabria, 2007) suggested the single PC lab for malware analysis, especially for researchers. Because deploying virtual lab environment on a single workstation or laptop using VMware or VirtualBox emulator is an easy solution.

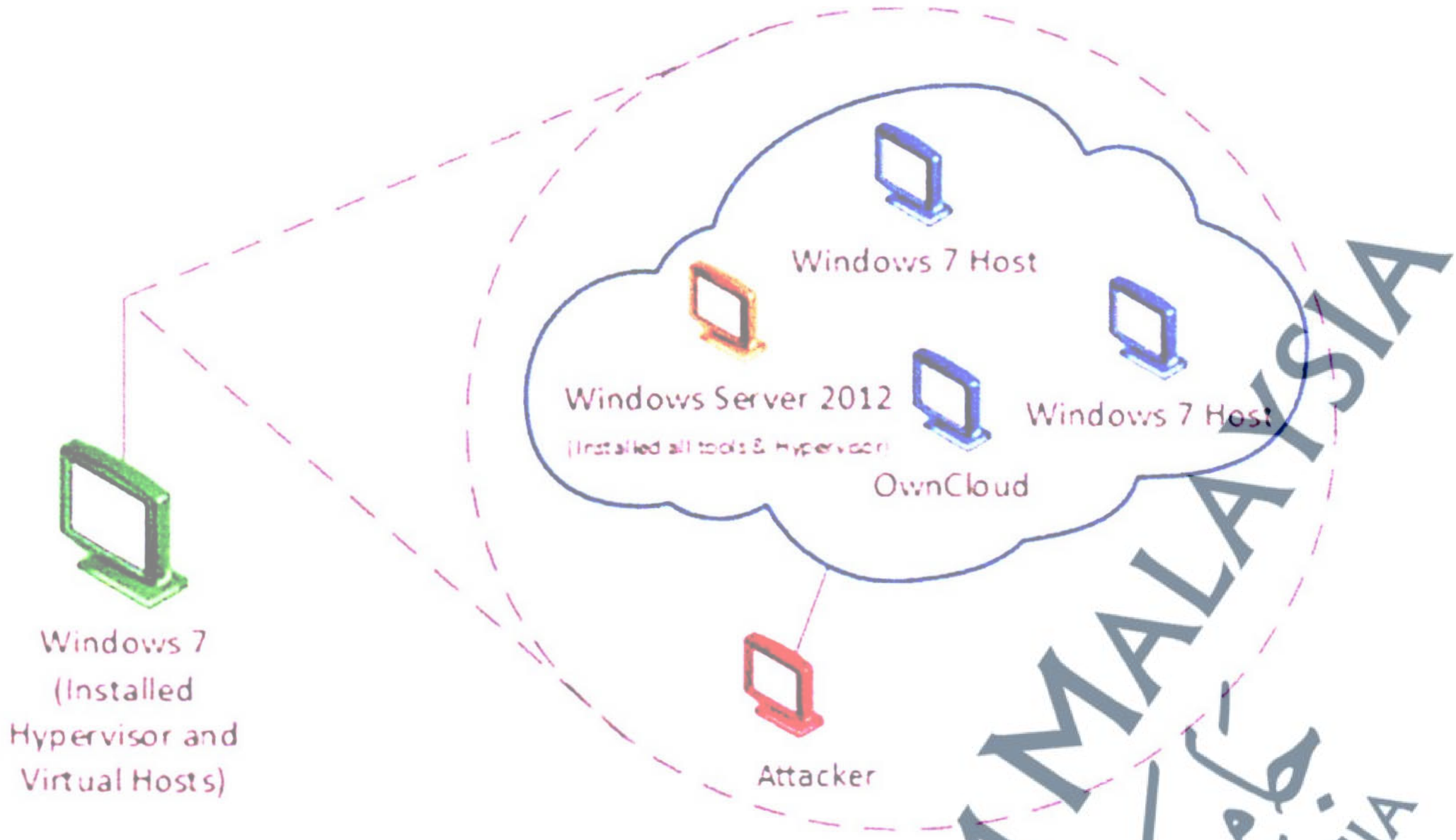


Figure 3.3: Controlled laboratory architecture.

Controlled lab architecture presented in Figure 3.3 is totally isolated from the internet. In a physical machine, cloud is deployed in the virtual environment. Attacker initiate occurrence to the cloud computing environment from outside of the cloud environment. All monitoring tools are installed in the server thereby monitoring cloud behaviour from inside the cloud environment.

3.3.3 Software and Hardware for Experiment

All experiments are performed on Intel® Core™ i5-3470 CPU and 4GB of RAM with 64-bit Windows 7 Home edition in a virtualised environment. VMware was used as virtualisation tools for creating VMs. A deep Freeze cloud tool (DeepFreeze, 2015) is used as reboot and restore technology protecting the host from worm infection. This tool helps to return the system to the original state once infected after being restarted.

Windows 7 was used as a host for the tools that were utilised for dynamic analysis in this research. In the initial stage of this research that commenced in 2013, Windows 7 was still the popular Operating System (OS) of its time. It was also known to be more stable than its predecessor. Windows 7 OS was more stable and never needed to

consume more RAM (Random Access Memory) in order to run. It was also more stable when running other software tools on it.

Therefore, Windows 7 was the chosen OS when this research began during the year of 2013. All dynamic tools that were used on the windows OS platform operated successfully without any serious issues.

The entire research performed by Abuzaid (2013), also made use of the windows 7 OS platform for the whole research analysis. Also, another researcher by the name of Saudi (2011), made use of the windows 7 OS platform to perform her entire analysis.

Various tools are used for monitoring file, process, port, network and registry in this work. List of tools for testing lab are presented in Table 3.3. VMware was used for visualisation and weka was used for testing and simulation of dataset carried out from the analysis. For cloud implementation ownCloud was used for its flexible API and security features.

Table 3.3: Software and Hardware Used In the Testing Lab Hosts

Tool	Purpose of Action
File Monitoring (Filemone)	To monitor all the actions associated with opening, reading, writing, closing and deleting files in relation to cloud worm.
Unpack tool	To decompress and unpack the cloud worm code within cloud server.
Process Monitoring (Preview v3.7.3.1)	To identify the resources used by all running processes, including DLLs and registry keys in cloud server.
Process Monitoring (Process Explorer)	Process explorer provides information on how the cloudworm is affected upon the victim computer.
Port Monitoring (PortMon)	To see which ports are listening on the trusted system. To record all TCP and UDP activity and to see various running programs that are sent or received from data to port on the cloud server.
Network Monitoring (Newt)	To look for backdoor listeners by which cloud worm attacks could be initited to be recognised byNeWT.

Network Monitoring (Wireshark)	To gather all traffic going to and from the target system, using a sniffer loaded on a system other than the victim computer within the cloud server.
Network Monitoring (Promiscdetect.exe)	To determine if the interface of victim's machine is in broadcast state mode, gathering packets destined for all systems on the LAN.
Network Monitoring (Nessus)	To monitor the listening ports.
Registry Monitoring (Regmon)	To display real time indication of all registry activity including creating, reading and writing registry keys.
Software for datatesting and Simulation (WEKA)	To perform data mining analysis and testing.
Virtual PC (VMware)	To build up virtual operating systems in a Computer.
Own Cloud	To conduct Cloud Server.
Deep Freeze Cloud Tool	To system restore on reboot.
Web based analysis tool	Virustotal.
PC-Win7	As Windows 7 due to its stability.

OwnCloud is a solution for cloud deployment with storage share and sync solution that could be hosted on its own servers. An open source community edition of ownCloud is available for cloud deployment. A universal file access front-end was provided by ownCloud to all of heterogeneous systems. Cloud resources could be accessed from anywhere, anytime on any device. It was also easier to control, manage and audit the resource sharing activity in order to ensure security and compliance measures are met (ownCloud, 2015). On the other hand, Hadoop is an open source framework for running applications and storing data on the cluster of commodity hardware which is especially developed for big data analysis (SaSHadoop, 2015) which is beyond one of the scope of this research. Hadoop allows distributed processing of large data sets across clusters of computers by using simple programming models (Hadoop, 2015). Hence, ownCloud was selected for cloud deployment over Hadoop.

The following steps are considered for loading Specimen:

- Before loading the specimen into the laboratory, the entire activities for the analysis must be observed thoroughly.

- The list of activities for the worm analysis in reference to Saudi (2011), are Monitor file activates, Monitoring processing, Monitor local network activity, Monitor registry activity and check registry changes, Monitor memory, Monitor network behaviour, Monitor tcp activities, Monitor udp activities, Monitor processes, Monitor all listening ports and Monitor dlls changes.
- The worm datasets were loaded into the testing computer using USB memory device.

3.3.4 Worm Analysis Process

Dynamic analysis processes was chosen to analyse downloaded worm samples. Virusshare.com malware report is also being followed during the analysis process. Virusshare.com (VirusShare, 2015) published worm analysis report through Virustotal.com which is implemented on cloud (Virustotal, 2015). Virustotal is a malware analysis system, completely based on open source solution which is totally customisable by the worm analyser. Virustotal has the ability to analyse any suspicious file in a matter of seconds by executing samples in an isolated environment. For the analysis of worm behaviour, software tools are installed in the cloud environment. Then sample worm are injected to the host to observe the behaviour. For observation, various monitoring tools are used to monitor files, network, tcp, udp, lan, all listening ports, registry, memory, processes and dlls. After getting monitoring results, obtained results were recorded for classified worms. Virustotal report is also used for verifying the analysis process which is an advance analysis tool. Worm analysis process is presented in Appendix A.

3.3.5 KDD Procedures

Knowledge Discovery in Databases (KDD) is an automatic, exploratory analysis and modeling of large data repositories. KDD is the organised process of identifying valid, novel, useful, and understandable patterns from large and complex data sets. Data Mining (DM) is the core of the KDD process, involving the inferring of algorithms that explore the data, develop the model and discover previously unknown patterns. The model is used for understanding phenomena from the data, analysis and

prediction (Giudici, 2010). Figure 3.4 illustrated the common KDD processes involved in developing knowledge.

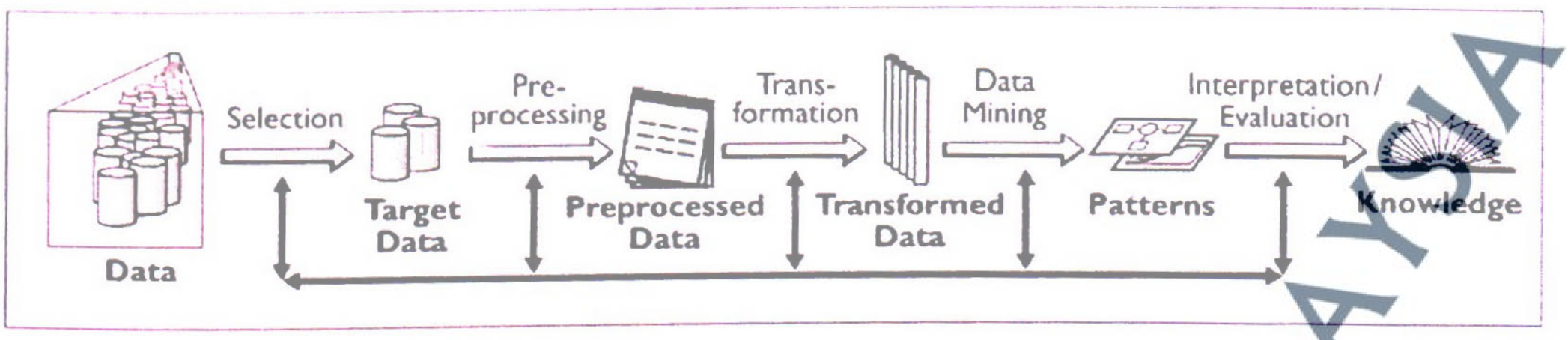


Figure 3.4: Steps of KDD process

The knowledge discovery process (Figure 3.4) is iterative and interactive, consisting of nine steps. Note that the process is iterative at each step, meaning that moving back to previous steps may be required. The process has many "artistic" aspects in the sense that one cannot present one formula or make a complete taxonomy for the right choices for each step and application type. Thus it is required to understand the process and the different needs and possibilities in each step.

Apart from various attack recognition techniques (Liao *et al.*, 2013; Lazarevic et al., 2005), data mining is one of the efficient pattern extractions for misuse detection by establishing profiles of normal networks for anomaly detection, and then generating a classifier to detect the attacks (Helali, 2010; Julisch, 2002). In past few years many researchers integrate KDD process for their research on intrusion detection system (Malyadri *et al.*, 2013; So-In *et al.*, 2014; Sahu *et al.*, 2015; Elekar *et al.*, 2015, Ferrand & Filiol, 2015) also used KDD for worm attack detection (Saudi, 2011; Nissim *et al.*, 2012; Kaur and Singh, 2014). For this research, KDD is used as a technique to identify the worm patterns in the datasets.

The purpose of data pre-processing process is to transform the cloud worm raw data into a more appropriate format for data extraction stage. This stage includes the feature selection, data cleansing to remove any noise, duplication and data transformation. The next stage is data mining process. The common technique implemented in data mining is classification in which the data pattern is extracted and achieved in this stage. The data is to be interpreted after the patterns have been extracted to make only useful information or knowledge kept for further analysis.

3.3.6 EGA KDD Process

In this research, enhancements have been carried out in the KDD data pre-processing and pattern extraction process which was achieved by data mining and classification. Under the data pre-processing activity, a dynamic analysis is implemented to obtain feature selection. While under the pattern extraction processes, statistical methods containing Chi-square and symmetric measure, security metrics and genetic algorithm are introduced. Chi-square and symmetric measure are used in order to determine the relationship between cloud worm characteristics and also to quantify the relationship strength. Meanwhile, security metrics is used to carry out the cloud worm threat level based on Confidentiality, Integrity and Availability (CIA). Also, genetic algorithm is also utilised for improvement of the accuracy detection rate of cloud worm. The total KDD process is presented and summarised in Figure3.5

In KDD process the target samples were selected from virusshare.com repository. Cloud worm dataset are then gathered and pre-processed by dynamic analysis. After dynamic analysis, the data sample is transformed into nominal data and then inserted into data mining for finding new pattern. Genetic Algorithm combines strongest and weakest worms. Using this combination, new types of worm characteristics could be generated. Through this, it could determine the types of worm which could be developed by the attacker in the future. So, worm detection accuracy is to be increased. Hence, this pattern has better accuracy rate when proposed GA is employed. Finally, new knowledge is achieved and also is to be able to carry out the further steps for the action and response.

In this work, feature was selected using dynamic analysis and then cleaning process is initiated to remove noise and duplication through data pre-processing. Data pre-processing takes raw data as input and transform raw data into appropriate new format. Next, statistical analysis was done using chi-square and symmetric measure to determine the relationship between worm characteristics. Security metric was defined for data mining process then, from the post-processing pattern of knowledge revealed.

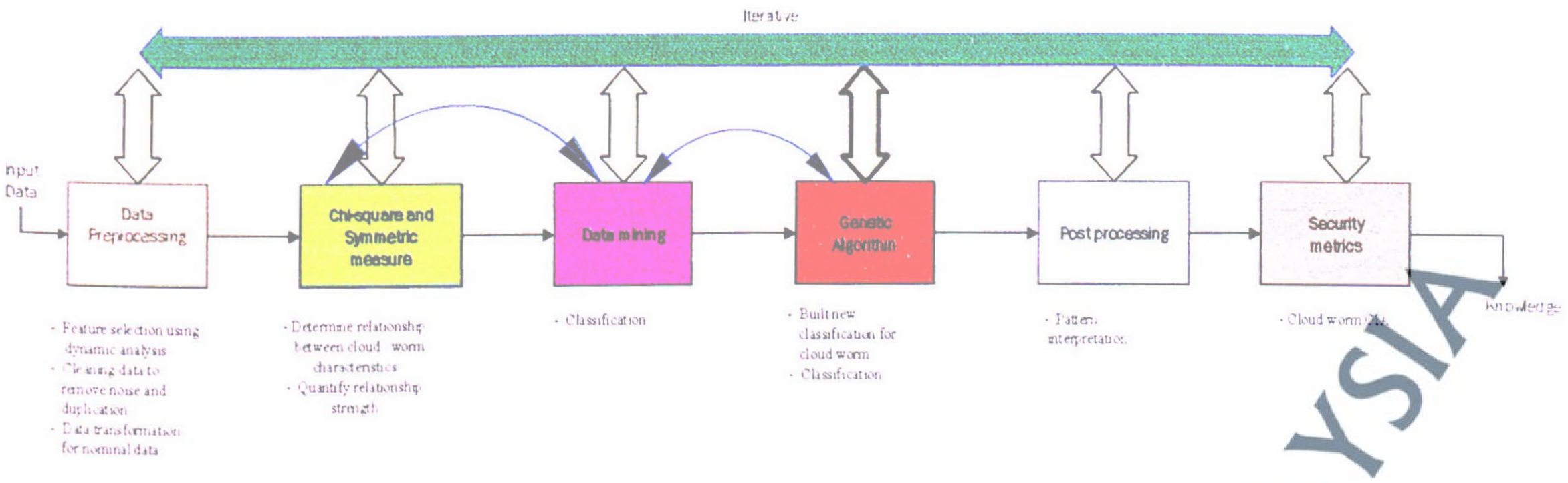


Figure 3.5: KDD process of EGA.

Pattern extraction was achieved by data mining and classification, which are known as the most common techniques. Types of algorithms implemented under classification depend on the goal that is wanted by the completion of the KDD processes. Valid extracted pattern from the data considered as knowledge or useful information is retained for future exploration.

The stages of EGA KDD processes are explained in the subsections below:

3.3.6.1 Data Pre Processing

The raw worm dataset is collected from virusshare malware repository. These dataset are transformed into new format according to worm characteristics for analysis. Feature selection initiated in this stage followed by data transformation and cleansing is undertaken for converting raw dataset into meaningful dataset. This process took about 40% time from the entire research process. Feature selection process is done by dynamic analysis in this stage. Complete description of each process in this stage is described in the rest of this section.

3.3.6.2 Feature Selection

The dataset collected from virusshare for this research contains 1018 worm samples. These samples are in various formats and needs to be converted into comprehensive format for analysis. Thus, feature selection is carried out through dynamic analysis. Feature selection was done in this research by a process which defined characteristics of every worm through dynamic analysis and tabulated into a meaningful dataset for

the further analysis. The chosen data in this thesis, as already defined under section 3.3.1, was analysed using dynamic analyses in a controlled lab environment which is referred to in section 3.3.2.

In this research, dynamic analysis procedure is used to complete the feature selection process. All findings and analysis are documented.

3.3.6.3 Dynamic Analysis

In dynamic analysis process, worm is activated in a controlled lab environment. In controlled lab, Windows 2012 server is installed as server operating system. Inside the windows server, hypervisor is installed including created multiple VMs with their own OS. Then owncloud is implemented for facilitated cloud platform. Then the necessary tools are to be installed for monitoring behaviour and finally injecting the cloud worm samples to monitor their activities. The behaviour and actions of the worm is observed and the characteristics are identified. Five different steps are involved in the procedure of dynamic analysis: monitoring file activities, process monitoring, network activities monitoring, monitoring system registry and complete analysis process. Dynamic analysis processes are illustrated in Figure 3.6 below.

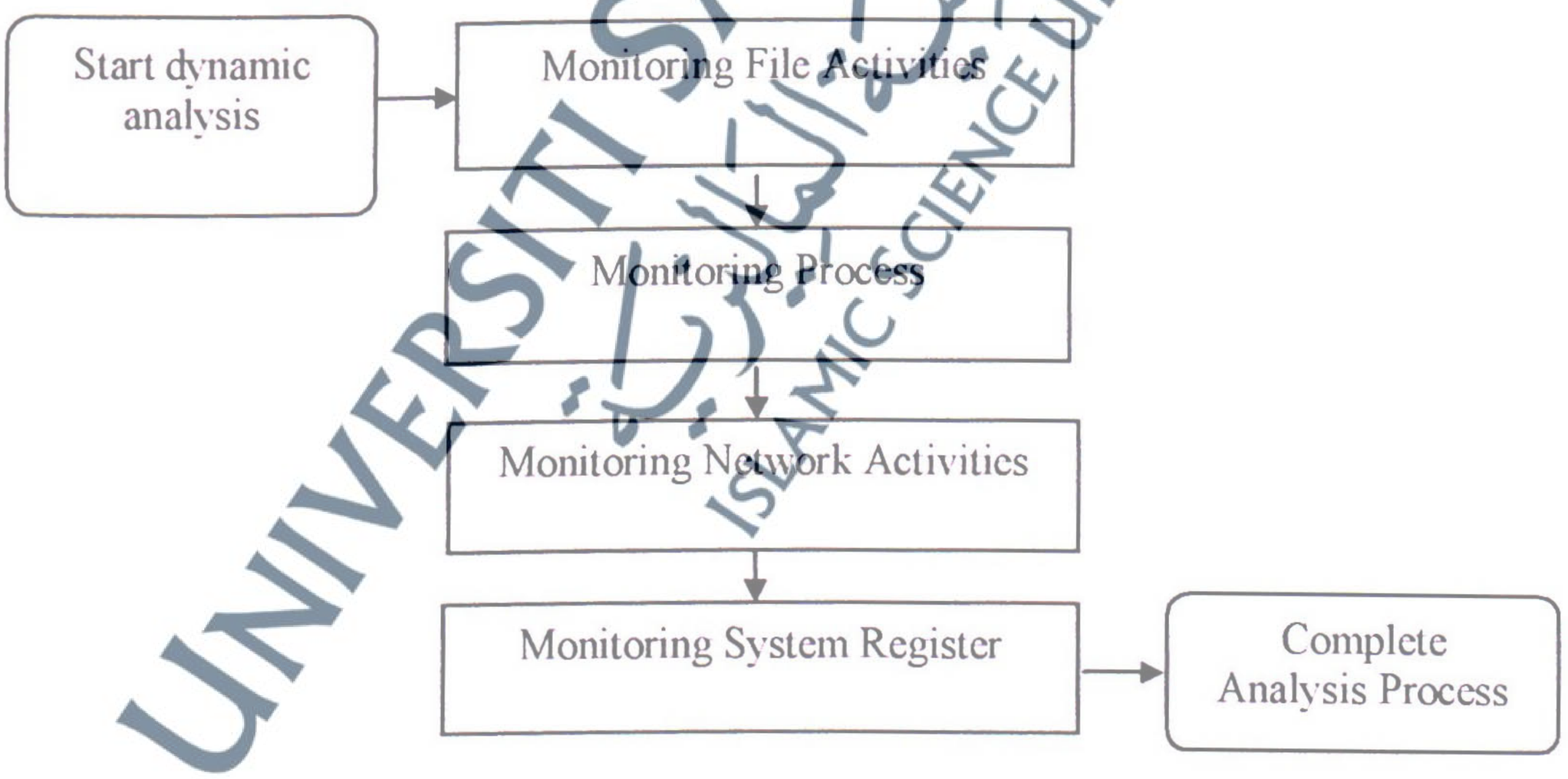


Figure 3.6: Dynamic analysis process.

For this research, the worm categorisation was done for the EGA in order to help increase the accuracy in worm detection rate in cloud. The worm categorisation was based on testing and comparison through dynamic analysis that is associated with other researchers' techniques such as those carried out by Saudi *et al.*, (2008); Pratama and Rafrastara, (2012); Abuzaid *et al.*, (2013); Rajesh *et al.*, (2015) and Suleiman and Husain, (2015). The categorisation or classification of the worm is related to five main attributes. These related attributes are infection, activation, payload, operating algorithm and propagation. Dynamic tools for the analysis were utilised in order to come up with the new categorisation. This categorisation is necessary due to the fact that the numerous types of worms are developed by different malware programmers. Because of these incidents, worms with different characteristics exist. This also became the reason why it is necessary to study more about worm characteristics especially in cloud computing. This categorisation can also be utilised for further research in the same filed.

The types of analysis that are frequently used by researchers in this field are static and dynamic analysis. For this research, dynamic analysis is the chosen technique due to the fact that the study and experiment to be performed will make use of virtual cloud process which also uses dynamic process instead of signature approach. In dynamic approach, the action and behaviour are usually studied because of the nature of the malware activities can be monitored and observed. It also has the capability to record changes in the malware that is made during execution. Also, it is evident that dynamic analysis can be viewed as a good source of information on malware behaviour. An accurate detection model can be obtained from dynamic analysis.

As for the signature approach, it is not going to be used in this research due to the fact that it is limited to static signature database. Static analysis may also fail in analysing unknown malware that uses code obfuscation techniques. Also, it needs to update the database regularly and human experts are required to develop the new signature of malware. Due to this reason, it can be understood that cloud computing utilises more of a dynamic approach not signature based approach because of its constant connectivity with internet. As it is based on signature, it only detects malware by comparing the pattern against database. For this reason dynamic analysis has been implemented for this research experiment.

3.3.6.4 Data Cleaning and Transformation

The data cleaning process is done during the collection of dataset from virusshare. Malware samples available in virusshare are identified by MD5 key and these key is checked for each sample to avoid duplication. This method ensures absence of noise, duplicates and outlierdata. However, noise, duplication and outlier data are checked after transformation. The pattern of worm characteristics are identified during dynamic analysis which are: infection, activation, payload, propagation and operating algorithm. Under these classifications, various sub classifications or attributes are identifies which are reported in chapter 4. Defining classification is an important part of this research because wrong identification of attribute might lead to less accurate results. Subsequently, these five worm characteristics were used by the experiments to represent all the dataset. Later, these dataset are used for statistical analysis in SPSS and also for data mining using WEKA. Characteristics of cloud worm dataset were transformed into nominal data with a numbering representation.

Furthermore, dataset used in this research from the Virusshare source consists of executable source code in the Windows PE format (i.e. .cpl, .exe, .dll, .ocx, .sys, .scr, and .drv) and some of them in web page programming, scripting and other programming language (i.e. .html, .pl, .sh, .c, .cpp, .java, .vbs). By executing worm the main features of the worm was extracted. Later, these features are transformed into an understandable format and used as an input for WEKA software. Virustotal is used for automated and advance dynamic analysis and manual dynamic analysis also done by using various tools.

Table 3.4: Data transformation examples

MD5 Key	Infection	Activation	Payload	Operating Algorithm	Propagation	New format of dataset
0c195a06c8f88ae025203aa7fedb06e0	Other users resource, hypervisor, VM image sharing, VM migration, VM rollback. VM isolation, communication and application	Human trigger, scheduled process and self-activation	Backdoor, DoS, Destructive,Steal information,Phishing and registry Shuffling	Stealth,Polymorphic and Anti-anti virus	Random	i23456789,a123,p123456,o123,P1
642876a70c89de2f31e1b983582a8709	Root privilege, hypervisor. VM image sharing. VM migration, VM rollback, VM isolation, communication and application	Human trigger. scheduled process and self-activation	Backdoor, DoS, Destructive, Steal information,Phishing and registry shuffling	Stealth, polymorphic and anti-anti virus	Random	i13456789,a123,p123456,o123,P1
0f31cc060476927061 51b8b08bd59774	Other users resource and application	Human trigger, scheduled process and self-activation	Destructive, Steal information, Phishing and registry shuffling	Stealth and polymorphic	No	i29,a123,p3456,o12,P0

From the worm source code, once it has been analysed using the dynamic analysis, the five main features of the cloud worm are being extracted into semi format structure comprising five different subareas to capture the worm characteristics. These five different subareas were transformed into nominal data with five numeric values which are used as the input to the machine learning algorithms, where the WEKA software is used. Worm sample characteristics are transformed into numeric representation then transformed into nominal data with various number representations. Weka only supports transformed nominal data for data mining. Few examples of transformed dataset are presented in Table 3.4. In this table, new format of first worm sample could be described as follows:

i23456789 represents Infection as - Other users resource, hypervisor, VM image sharing,VM migration, VM rollback, VM isolation, communication and application.

a123 represents Activation as - Human trigger, Scheduled process and Self-activation

p123456 represents Payload as - Backdoor,DoS,Destructive,Steal information, Phishing, and registry shuffling.

o123 represents Operating Algorithm as – Stealth, Polymorphic and Anti anti-virus.

P1 represents Propagation as - Random

Transformed dataset used as an input in weka is shown in Figure 3.7.

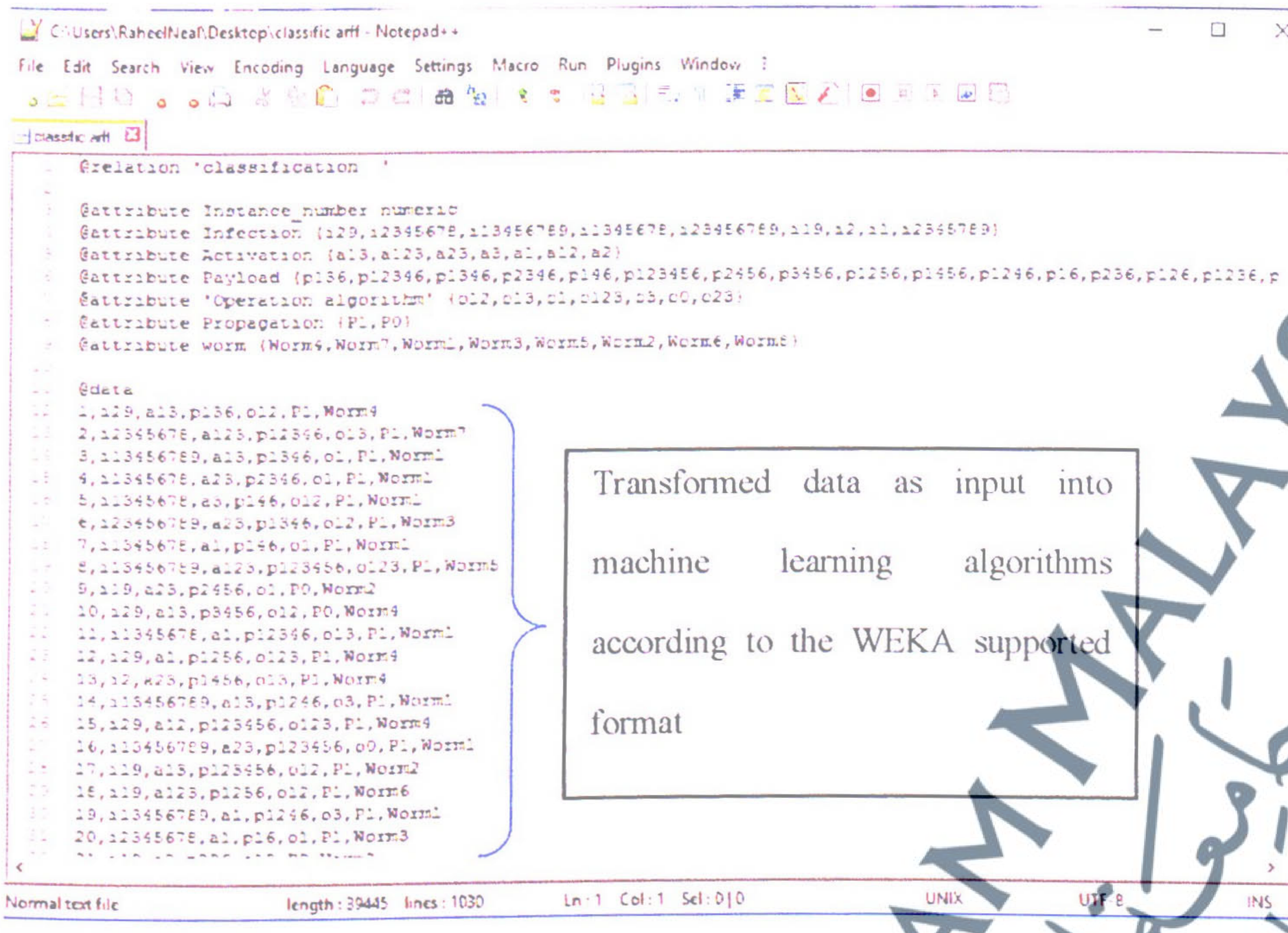


Figure 3.7: Transformed dataset used in weka.

The classification formation of the new EGA cloud worm detection model and result of dynamic analysis are explained in details in chapter 4.

3.3.6.5 Independence Test of cloud worm dataset

A Chi-Square test of independence was performed for each feature to determine if a relationship exists between the feature and the target variable. The term document matrix was transformed to a binary representation to get a 2-way contingency table.To prove the relationship between EGA cloud worm classifications, statistical analysis was accompanied. From five main features: infection, activation, operating algorithm, payload and propagation were evaluated. If the expected frequency is less than 5, it is the standard and all researchers use this value for their study. Assumption about the shape of the underlying distribution is not required in nonparametric tests. In this test, it is assumed that data is revealed from random samples. For each category the expected frequencies must be no less than 1. Less than 20% of the categories would have expected frequencies of less than 5. On other meaning, because this is Statistical prerequisite to test the chi-square, it means there are at least 80% of the actual data

already available even subject to the chi square test. The dataset was a nominal data which is also known as categorical data set and testing was done based on frequencies. After that, the dataset was converted into percentage for further analysis. The statistical analysis was conducted using the Software SPSS.

Statistical analysis was conducted once the data pre-processing method is completed to analyse the datasets. According to Giudici (2010) the statistical analysis provides added value to the analysis by data mining. In this research, the Chi-square statistical test, symmetric measure, training and testing validation under data mining was implemented.

Chi-square and symmetric measure tests were used to test the relevance of the EGA worm classification and the EGA relational model. These tests were undertaken to determine the relationship whether it exists between worm characteristics chosen in the EGA relational technique, followed by the symmetric measure to quantify the strength of the relationship.

Chi-Square test compares actual frequencies with the expected frequencies statistically by cross tabulation to verify that the result happens by chance or not (Greasley, 2007). It is also able to measure the variance between the experimental results cell count and what to expect, whether the columns and rows were distinct. The formula used by Chi-Square on these data is shown in equation (3.1)

$$X^2 = \sum \frac{(O-E)^2}{E} \qquad (3.1)$$

In equation (3.1), O represents the observed frequency, E represents expected frequency, and X^2 denotes Chi-Square. The value of X^2 cannot be assessed except the number of degrees of freedom (df) associated with it is known. The number of df associated with any X^2 may be computed easily.

If data is distributed randomly, then those estimated are known as expected frequencies. Under null hypothesis average count, one would guess is the expected count in that cell. Generally, each cell of the contingency table expected count is calculated using equation 3.2.

$$\frac{row\ total * column\ total}{grand\ total} \tag{3.2}$$

The Chi-square and symmetric measures involve null hypothesis (H_0) and alternate hypothesis (H_a) testing. If the null hypothesis (H_0) is accepted, then it is proved that there is no significant difference between observed and expected frequencies. On the other hand, if the null hypothesis (H_0) is rejected then the alternate hypothesis (H_a) is accepted and then it is proved that there is a relationship between observed and expected frequencies. The confidence level of significance is chosen as 95%, thus the P value is set to 0.05, meaning less than 5 from 100 happens by chance. The result is obtained from statistical tests, presented in the chapter 4. An analysis of achieved results is also presented in the same chapter.

Fisher's exact test (Conover, 1999) is another statistical test used to determine whether a non random association exists or not. But fisher's exact test is most commonly functional with the 2×2 matrices, and is computationally unwieldy for large matrices. The statistical null hypothesis for the test is that there is no association between two discrete-values. The Fisher`s exact test has the equal objective as the Chi-square test. However, it is limited to the expected counts of not more than five. In Chi-square test, the expected counts must be more than five (Saudi, 2011). Because this is Statistical prerequisite to undergo the chi-square distribution, that is if the expected count less than 5, It means that the significant data did not get it (missing) in the sense that there are more than 20% of the data missing and therefore it cannot rule on the approval or rejection of the hypothesis. Data used in this research is nominal data so; it can be concise that the Chi-square test and symmetric measure find the relationship between worm characteristics in cloud. To quantify the strength of the relationship, EGA model is applied for worm detection and behaviour analysis within the cloud.

3.3.6.6 Security Metrics

This research used two important measures, weight and severity which were calculated using security metrics in order to conduct an in-depth study. Explanation of security metrics to determine weight and severity values is discussed in Chapter 5.

Security metrics is a method that helps to measure, quantify and classify security based information. Threats defined by the experiments and analysis are transformed to the metrics so that threat level could be easily identified and measured. Following these metrics, it is tranquil to identify and understand security holes, problems, flaws, weaknesses or the harm they can initiate to the security infrastructure. It also helps to investigate the present countermeasure process performance and recommend the improvement of any countermeasure process or technology if necessary (Jaquith, 2007).

Security metrics is applied to find the level of threat meaning and how it can harm the system. From this, threat level is rated by high, medium and low keyword. This level is defined by considering worm attributes in the cloud with various rules. These rules indicated how cloud worm could damage system if not put in any protection against worms in cloud. The security metrics procedures are already being employed in EGAKDD Processes for worm detection as presented in Table 3.5.

Table 3.5: EGA process security metrics.

Security metrics processes	**Applying security metrics in EGA**
1) Define cloud worm threats	Yes
2) Represents cloud worm threats into metrics	Yes. – Worm data was represented based on payload, infection, activation, propagation and operating algorithm. – Formation of the EGA worm classification and EGArelational model.
3) Understand and identify the vulnerability, flaw, problem, weakness and damage to security Infrastructure	Yes. – Run the dynamic analysis. – Identify the need to assign weight and severity value to assign the countermeasure process.

An in-depth study and understanding of worm behaviour and architecture is crucial for EGA research. This work leads to the development of EGA cloud worm classification and the EGA cloud worm relational model. Primarily, cloud worm characteristics need to be observed and defined. Then, by the support of dynamic analysis, the worms are analysed and simplified into cloud worm representation, which comprises payload, activation, operating algorithm, infection and propagation.

A thorough analysis related to security holes, problems, flaws, weaknesses or the harm they can initiate to the security infrastructure is monitored closely. As a result, weight and severity are chosen as two main attributes in assigning the countermeasure process. It is suggested that, all worms with a high severity level need to be isolated. These measurements were already taken into consideration when the cloud worm analysis was conducted. Therefore, as a result, the weight and severity performance and value are tested based on data criticality level, infrastructure availability and loss of productivity.

3.3.6.7 Data Mining

Classification is used in data mining to find hidden patterns. Classification is a data mining function that assigns items in a collection to target categories or classes. The goal of classification is to accurately predict the target class for each case in the data. In this research, classification technique has been applied. However, all datasets used in this research are nominal data.

3.3.6.8 Classification

In order to test the accuracy of the eight different types of cloud worm assignment, the classification algorithms are integrated (the findings can be examined in Chapter 5). The classification algorithms chosen are the Naïve Bayes, Decision Tree (J48) and K-nearest Neighbours (IBk). These algorithms were chosen by various works for malware detection (Dai *et al.*, 2009; Siddiqui *et al.*, 2009; Saudi, 2011). However, their work is related to the computer network while, cloud is considered as a types of complex network system. For this reason, these algorithms were chosen for this research.

These classifications are applied so that a comparison can be made between them, which therefore, enable identification of the most accurate classification algorithm.

3.3.6.9 Data Post-processing

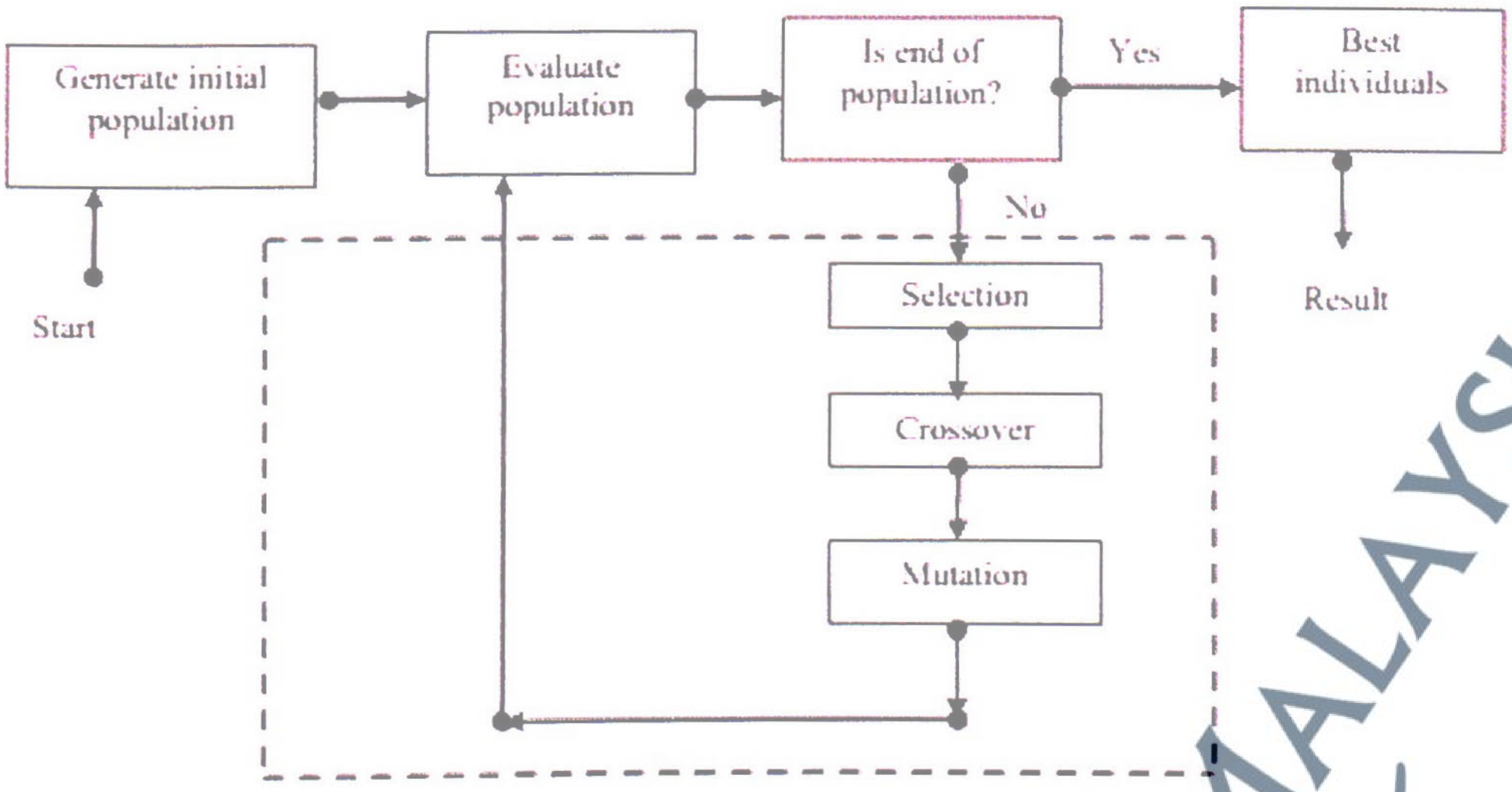
At this stage, the complete pattern extracted from the data is interpreted so that useful knowledge is produced by the end of all the processes. Later, the pattern extracted can be simplified using graphs or any suitable methods to represent the complete extracted pattern for any further exploration or analysis. In EGA, at this point, a conclusion and summary can be made based on all the findings to ensure all the objectives for this research are achieved successfully.

3.3.6.10 Genetic Algorithm for Cloud Worm Detection

Genetic algorithms (GA) are search algorithms based on the principles of natural selection and genetics by Darwin's, introduced by J Holland in the 1970s and inspired by the biological evolution of living beings. Genetic algorithms explore the fittest individual by producing generations iteratively. The generation of new offsprings includes the operations such as crossover, mutation and selection operations (Golberg, 1989).

GA evolves a population of initial individuals to a population of high quality individuals, where each individual represents a solution of the problem to be solved. Each individual is called chromosome, and is composed of a predetermined number of genes. The quality of each rule is measured by a fitness function as the quantitative representation of each rule's adaptation to a certain environment. The procedure starts from an initial population of randomly generated individuals. Then the population is evolved for a number of generations while gradually improving the qualities of the individuals in the sense of increasing the fitness value as the measure of quality. During each generation, three basic genetic operators are sequentially applied to each individual with certain probabilities, i.e. selection, crossover and mutation.

The algorithm flow is presented in Figure 3.8 (Dhopte *et al.*, 2014). GA can produce fittest generation for the future survival, due to this it can be used in cloud worm detection to detect known and unknown threat by using GA nature. This research use GA to improve detection accuracy in cloud worm detection.



Source :(Dhopte et al., 2014)

Figure 3.8: Process flow of GA algorithm

In this research, selection, a crossover and mutation process is initiated to improve detection accuracy and determine unknown cloud worm characteristics to detect future attack. Under the selection process, this research has been used for the Selection Proportional of Fitness technique by combining strongest and weakest cloud worm to generate new types of cloud worm characteristics. As for crossover, the Tree Crossover technique has been used to select sub tree at random from each parent cloud worm. The parent cloud worm has been combined to generate new cloud worm characteristics.Under mutation process, the Tree Mutation has been used to give a new cloud worm with new characteristics which is different from the parents. Additionally, Evolution controller process has been introduced to ensure better new generation for the future. Following OlexGA, the propsed EGA is implemented. Detailed explanation can be found in chapter 5.

3.3.7 Experimental Evaluation

The evaluation of cloud worm detection was done in a controlled lab environment by integrating dynamic analysis for monitoring behaviour, statistical analysis for finding relationship between features and data mining to find detection accuracy rate.

A contingency table which is also known as confusion matrix is used to describe experimental results. Confusion matrix represents predicted and actual classifications (Kohavi & Provost, 1998). Confusion matrix dimension denotes in *m x m* where *m* is the number of various label values. *2 X 2* confusion matrix is shown in the Figure 3.9.

		Predicted class	
		Yes	No
Actual class	Yes	*true positive (TP)*	*false negative(FN)*
	No	*false positive (FP)*	*true negative(TN)*

Figure 3.9: Examples of *2 X 2* confusion matrix

3.3.8 Performance Criteria

All criteria for performance measurement are described in this section. These criteria are used to measure performance throughout this research. True positive (TP), false positive (FP), true negative (TN) and false negative (FN) rate was measured. Other parameter such as: F-Measure, accuracy, error rate and precision are also measured.

The TP occurs when data is correctly classified as class A while FP means the data is being misclassified as class A. TN occurs when data is correctly classified wrongly in class A and FN occurs when the data is wrongly classified as a different class. Rate of these four parameters is calculated using Equation (3.4), (3.5), (3.6) and (3.7)

$$True\ positive\ rate\ (TPR) = \frac{TP}{(TP+FN)} \tag{3.3}$$

$$False\ positive\ rate\ (FPR) = \frac{FP}{(FP+TN)} \tag{3.4}$$

$$True\ negative\ rate\ (TNR) = \frac{TN}{(TN+FP)} \tag{3.5}$$

$$False\ negative\ rate\ (FNR) = \frac{FN}{(FN+TP)} \tag{3.6}$$

F-measure is a way of combining precision and recall scores into a single measure of performance the Equation (3.8) shows the measurement process of f-measure.

$$F - Measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \qquad (3.7)$$

Precision is the proportion of relevant documents in the results returned and Recall is the ratio of relevant samples found in the search result to the total of all relevant samples. The higher the Precision and Recall values mean, the more relevant samples are returned more quickly. Precision is calculated using Equation (3.9).

$$Precision = \frac{TP}{(TP+FP)} \qquad (3.8)$$

Accuracy referred as the value of correct classification. So the correct classifications are the TP and TN. The accuracy calculated by dividing the addition of TP and TN by the addition of TN, TN, FP and FN as shown in Equation (3.10).

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \qquad (3.9)$$

On the other hand, error rate is calculated by subtracting accuracy from 1 which represents the error of the classifications as presented in Equation (3.11).

$$Error\ rate\ = 1 - Accuracy \qquad (3.10)$$

3.4 Experimental Procedures

Cloud worm dataset are collected from virushare.com, a detail explanation about these dataset are presented in section 3.3.1. The proposed EGA technique and algorithms are tested based on this collected dataset. During the dataset collection and cloud worm sample characteristics evaluation phase, many samples were included and excluded based on their characteristics. Finally, 1018 cloud worm out of 1195 samples are selected for the experiment. Each cloud worm sample tested individually in a controlled environment revealed their characteristics. Some cloud worms are in dormant phase based on their activation characteristics. The worms which are activated by human triggering are easier to evaluate by the software tool. For the other two types of cloud worm which are activated by self activation and scheduled process, they are trickier to evaluate in their characteristics. It was found that these kinds of

cloud worm can easily be analysed by the virustotal tool which generate a report for each worm and which is cloud based analysis. Virustotaltool manipulate worm in such a way that they activate themselves while they are self activated and have scheduled process characteristics. Hence, this tool was used to tackle this issue especially for the cloud worm which are generally in dormant phase. The activity of a worm is defined by the software tool and virus total report. When a sample worm is executed in a controlled environment, the activity of the worm is triggered by software tools and considered as abnormal activity. Similarly, virustotal report also represents the abnormal activity of a worm. So, the baseline for the abnormal activity is the malicious behaviour detected by the dynamic analysis tools and virustotal tool. Every software analysis tool produces many sub features. However, this work is focused on the features based on the proposed classification. For example, infection could be done by communication and application. If it is done by communication, network monitoring dynamic analysis tools are able to figure out whether worm is trying to scan for remote port to initiate attack or not. Similarly, the file monitoring tools can find when a worm is trying to bind itself with an application. In this way, feature and sub features are influenced by the cloud worm activity. Based on the proposed characteristics of each worm categorised, If the characteristics are found in the worm then the value for that category is to be yes and if not found then it is no. More detail about each characteristic is to be found in section 4.3.1. After finding the characteristics of all cloud worm samples, the dataset is transformed into nominal data for further analysis. Then, independence test is to be undertaken to find the relationship among cloud worm characteristics. Chi-square and symmetric measures testing was initiated on obtained dataset. Frequency analysis helps to identify most important cloud worm detection attributes and also helps to determine the relationship between the attributes. Experimental results for frequency analysis are presented in section 4.5. After frequency analysis, a supervised learning is used to find the malicious and benign among worm samples. Hence, classification is used to find the pattern from the worm samples. Here, benign represents the samples which are not malicious program which does not cause any major harm to the system. As an example, a worm just replicates itself steadily, but does not delete or alter anything in the system. In this case worm is too small in size and the steadiness replication characteristic does not affect the memory much. The dataset obtained was labelled.

The dataset consist of different types of worms. A new EGA algorithm is proposed and classification is carried out to find whether proposed algorithm can classify worm properly or not. Finally, security metrics are proposed and again tested with proposed EGA classifier. The flow chart for the experiment can be seen in Figure 3.10 below.

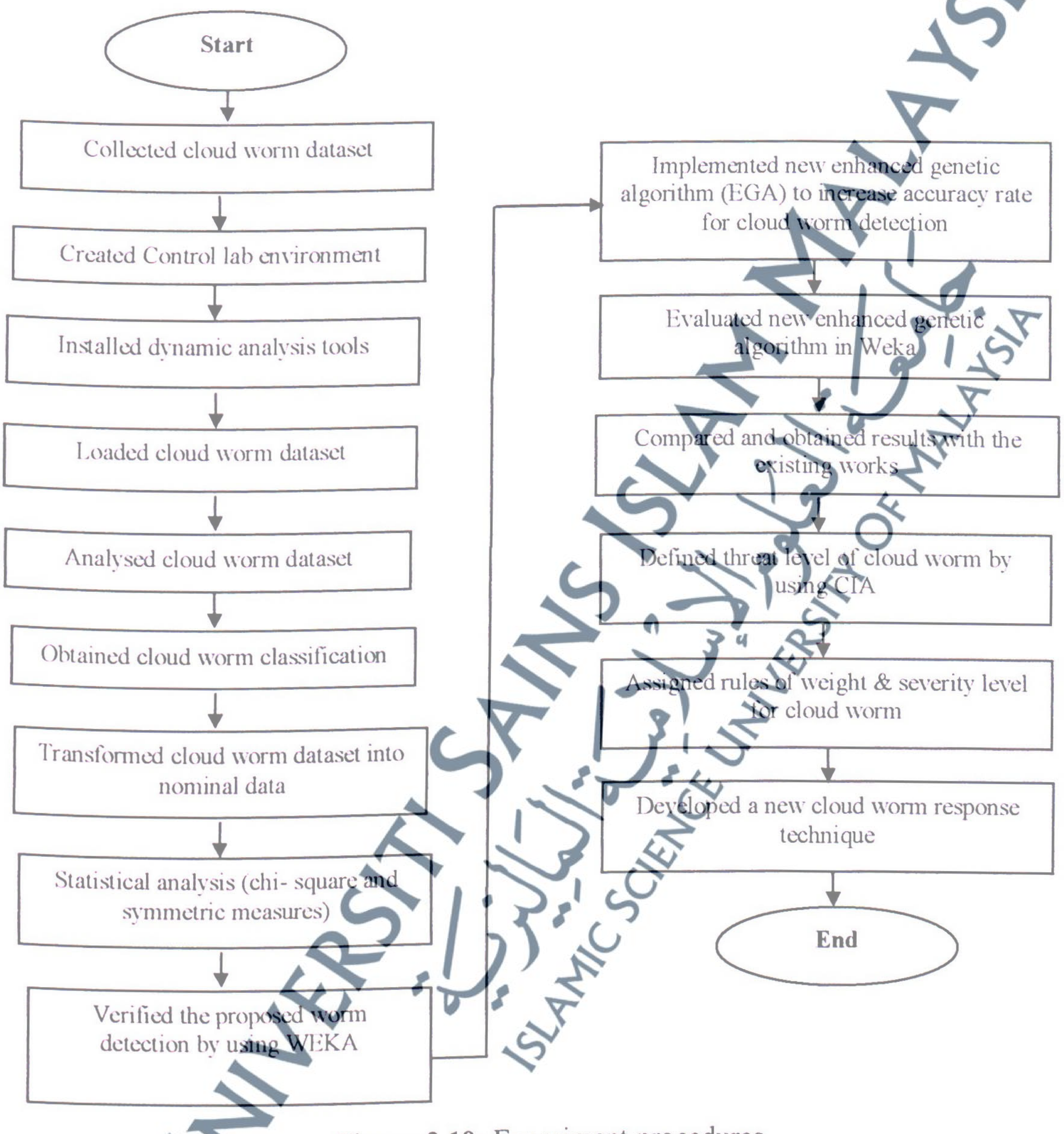


Figure 3.10: Experiment procedures

A brief description of each phase above can be seen at the Table 3.6.

Table 3.6: Brief Descriptions of Experiment Procedures

Phases	Description
1. Collected cloud worm dataset	Two type of dataset were collected for this research which are worms and benign dataset. The dataset is used to build up a detection technique. For both malicious and benign 1195 samples were obtained.
2. Created Control lab environment	To carry out research experiment by installing dynamic analysis tools in VMware to perform analysis on the cloud worm dataset in order to produce a new cloud worm classification.
3. Installed dynamic analysis tools	Installed dynamic analysis tools such as file monitoring, process monitoring and network monitoring in order to monitor and observe the cloud worm activity.
4. Loaded cloud worm dataset	The cloud worm dataset were loaded into the control lab testing using USB memory device and injected the cloud worm into control lab environment.
5. Analysed cloud worm dataset	Cloud worm was activated and analysed in a controlled lab environment. Then, behaviours and actions of the worms were observed and the characteristics were identified.
6. Obtained cloud worm classification	The results were recorded for classified cloud worms based on infection, activation, payload, operating algorithm and propagation.
7. Transformed cloud worm dataset into nominal data	After dynamic analysis was completed, characteristics of cloud worm dataset were transformed into nominal within a numbering representation data and then inserted into data mining for finding new pattern by using WEKA.
8. Statistical analysis (chi-square and symmetric measures)	Statistical analysis was done using chi-square and symmetric measure to determine the relationship between worm characteristics and to quantify the strength of the relationship and also to verify that the result happens by chance or not by using SPSS software.
9. Verified the proposed worm detection by using WEKA	By using different classification algorithms which are built within weka such as IBK, J48 and Naïve Bayes in order to identify most accurate classification algorithms.

10. Implemented new enhanced genetic algorithm (EGA) to increase accuracy rate for cloud worm detection	To implement EGA technique, OlexGA was first implemented and embedded in weka. Through OlexGA, enhancement was made using selection of proportional of fitness, Tree crossover, Tree mutation techniques and a new technique called Evolution Controller in order to increase accuracy rate for cloud worm detection
11. Evaluated new enhanced genetic algorithm in WEKA	To evaluate EGA in weka to obtain the required output.
12. Compared and obtained results with the existing works	To compare the experimental results with existing work and other established algorithm through evaluation criteria such as measuring the accuracy detection rate.
13. Defined threat level of cloud worm by using CIA	To define the threat level of damage by the cloud worm using Confidentiality, Integrity and Availability (CIA) as used by Swanson and Gregg (Swanson, 2001; Gregg, 2005).
14. Assigned rules of weight & severity level for cloud worm	To define the rules to obtain the weight & severity level for cloud worm. During the process CIA, rules are defined to get the value of weight considering the threat level of five main features which are infection, activation, payload, propagation and operating algorithm. Based on CIA, weight value is defined. Weight and severity values are justified by the CIA. Based on the weight value, severity value also defined.
15. Developed a new cloud worm response technique.	To propose new algorithm for cloud worm response. It is a new algorithm after the detection process. There was no previous study which has been performed in relation to worm response in cloud environment. Hence, the proposed algorithm was claimed as new algorithm in cloud worm response domain.

3.5 Summary

This chapter discussed the methodology of whole research processes to propose a technique for cloud worm detection and response. Stated methodology is the main column of this research which provides detail guidance about research process, experiments and results analysis. It ensures that a consistent and reproducible approach is used from the first activity of the research processes until all the processes are complete. Details of how these research processes are applied and the findings can be found in Chapter 4, Chapter 5 and Chapter 6.