CHAPTER 4: EGA CLOUD WORM CLASSIFICATION

4.1 Introduction

Worm has become a real threat to many organisations and computer users for more than a decade. According to Cyber Security Malaysia, four types of major security incidents are often reported which are; fraud, spam, intrusion, and malicious codes (MyCERT, 2015). Malicious codes attack the systems and applications by changing their original codes, causing these systems and applications to malfunction or unable to be used for normal operation. These malicious programs can replicate by themselves, and they have the ability to automatically transfer from one infected machine to another vulnerable machine via a network without the owner's consent. Therefore, worm is considered as one of the most dangerous types of malicious codes, which attacks huge volume of Internet applications through software codes.

Recently, worm has significantly increased its negative influence and has created a drastic chaos in the world of computers and the cloud computing environment. In terms of cloud, worm intrudes and tries to damage a service or application in cloud structure and puts itself as an authorised user to implement its malicious code into cloud structure. As for cloud worm detection, classification is one of the decisive processes that must be considered in order to ensure the effectiveness of the detection process. Classification method has been in large degree, used in malicious code analysis specifically in measuring the efficiency of detection for a new or unknown sample of malicious code. Cloud worm classification can be used as the basis for cloud worm detection and also to help increase the accuracy rate compared with other existing methods.

In this chapter, a worm classification in cloud computing environment is developed as a part and basis of a new worm detection technique in cloud computing environment. The technique is based on the proposed classification especially for worm attack in cloud. The proposed technique is evaluated using statistical methods and data mining techniques. The proposed techniques by the various researchers are mostly based on

theoretical model. However, some research initiatives addressed dynamic behaviour analysis but these works are ongoing. The research approach of this study is based on dynamic behaviour analysis and genetic algorithm.

4.2 Related Works

Worm classification was carried out through different categories which are all based on behaviour. The classification consists of email worms, polymorphic worms, stealth worms and file worms (Rajesh *et al.*, 2015). A study by Saudi (2011) also stated that worm classification can be utilised as a basis for a worm detection and response technique, and the author also classified the worm into different categories. The study on worm classification has been performed based on the testing and comparison associated with the research by Dabirsiaghi (2008), Nazario *et al.*, (2001), Helenius (2002), Skoudis and Zelster (2004), and Saudi *et al.*, (2008).

Based on these worm's basic features using dynamic analysis tools and experimentation, a new cloud worm classification is to be introduced in this research. Cloud worm classification can be utilised as a basis for worm detection and response method which helps increase the accuracy detection rate. More detailed classification can be referred in section 4.3.1.

Different researchers have provided description about the impact of worm attacks in cloud. However, some researchers did classification of worm and malware but not in cloud. This study is specifically concentrated on worm attacks in cloud which forms the basis for conducting this research.

4.3 EGA Technique for Worm Detection in Cloud

In this section, a new technique known as Enhanced Genetic Algorithm (EGA) is proposed. This technique represents the steps to cloud worm detection and response. The proposed model works in four steps as shown in Figure 4.1. Initially, sample is collected and then classified using the proposed EGA worm classification. Next, worm characteristics are analysed and statistical analysis is performed to find the

relationship among the sub features. In the next step, checking the detection accuracy is carried out using various in-built classifiers available in weka data mining tool. The existing GA algorithm can be added in this stage if available. This existing algorithm helps in benchmarking the proposed algorithm. Finally, a new enhanced GA could be proposed and tested using weka data mining tool. Based on the results, it could be suggested on how it responds to cloud worm after detection.

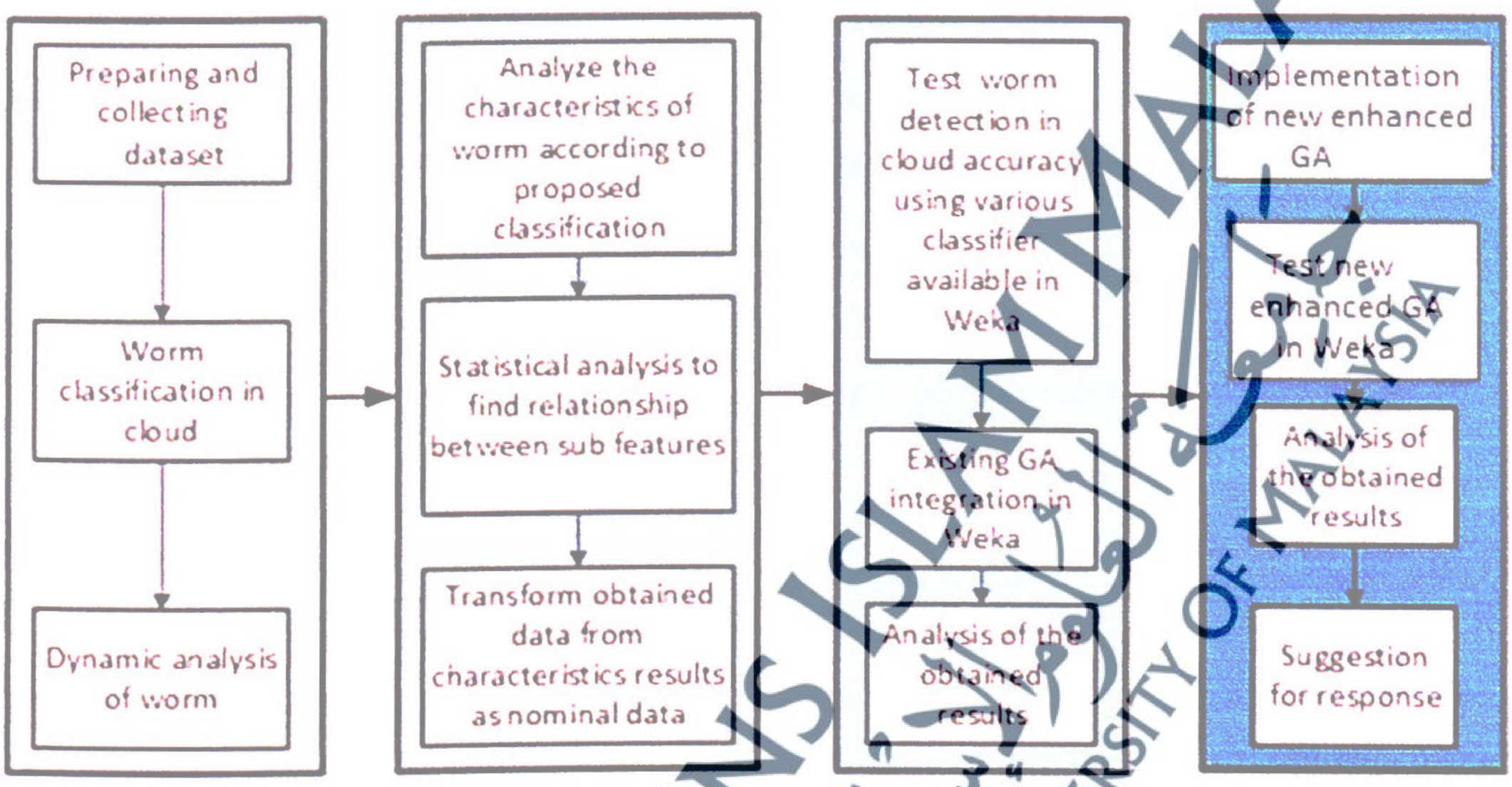

Figure 4.1: EGA technique for cloud worm detection and response

Cloud worm classification, statistical analysis and data mining results are elaborated and discussed in the following sections. Experimental results from data analysis and data mining are summarised, and brief analysis of the results is also presented in this chapter. Beside other established algorithms, a Genetic Algorithm (GA) for data mining is used for cloud worm classification, and this method was proposed by Pietramala *et al.*, (2008). The proposed GA was integrated in WEKA in the experimental environment.

The motivation to use this technique is the attainment of improved level of accuracy rate in the detection of worm activities in cloud. Due to this reason, worms are more challenging and sophisticated in order to reduce worm cloud infection. This GA model could be developed repeatedly by new researchers in this area. The proposed technique could be used in other cloud threat as well as in malware detection in cloud.

4.3.1 EGA Cloud Worm Classification

A new worm classification in cloud computing environment has been developed, based on the experiment conducted in controlled laboratory environment. According to Weaver *et al.*, (2003), worm has five basic characteristics known as infection, activation, payload, propagation, and operating algorithm. Infection refers to the process of how a computer becomes infected by a worm while activation is denoted as the process of triggering mechanism of a worm by which it tries to enter the host. The destructive mechanism of a worm is known as payload whereas propagation is the characteristics of a worm on how they spread out and find new hosts to infect. Meanwhile, operating algorithm defines how worm can avoid from being detected as malicious code. Although worm has those basic characteristics, each of these characteristics can be varied based on the network environment and domain.

Regarding the experimentation and dynamic analysis that is going to be conducted, it is recommended that a well-structured worm cloud classification, bearing in mind the characteristics in worm, can be used as the basis for a worm detection and response technique. Therefore, an Enhanced Genetic Algorithm (EGA) worm classification is produced by testing and comparison associated with the research by Rajesh *et al.*, (2015), Abuzaid *et al.*, (2013), Suleiman and Husain (2015), Saudi *et al.*, (2008), and Pratama and Rafrastara (2012). EGA worm classification is referenced in accordance with five main attributes. The attributes are infection, activation, payload, operating algorithm and propagation. Figure 4.2 illustrates the complete EGA classification for worm attack in cloud. EGA worm classification used in this thesis helps to increase the accuracy rate compared with the existing methods which will be obtained from the experimental results, where detection is explained in detail in Chapter 5 (section 5.7). In this experiment, dynamic and Virustotal web based tools were used to observe the injected worm into the virtual cloud. The findings using the dynamic tools including worm's sub features which led to new classification are shown in Figure 4.2 below.

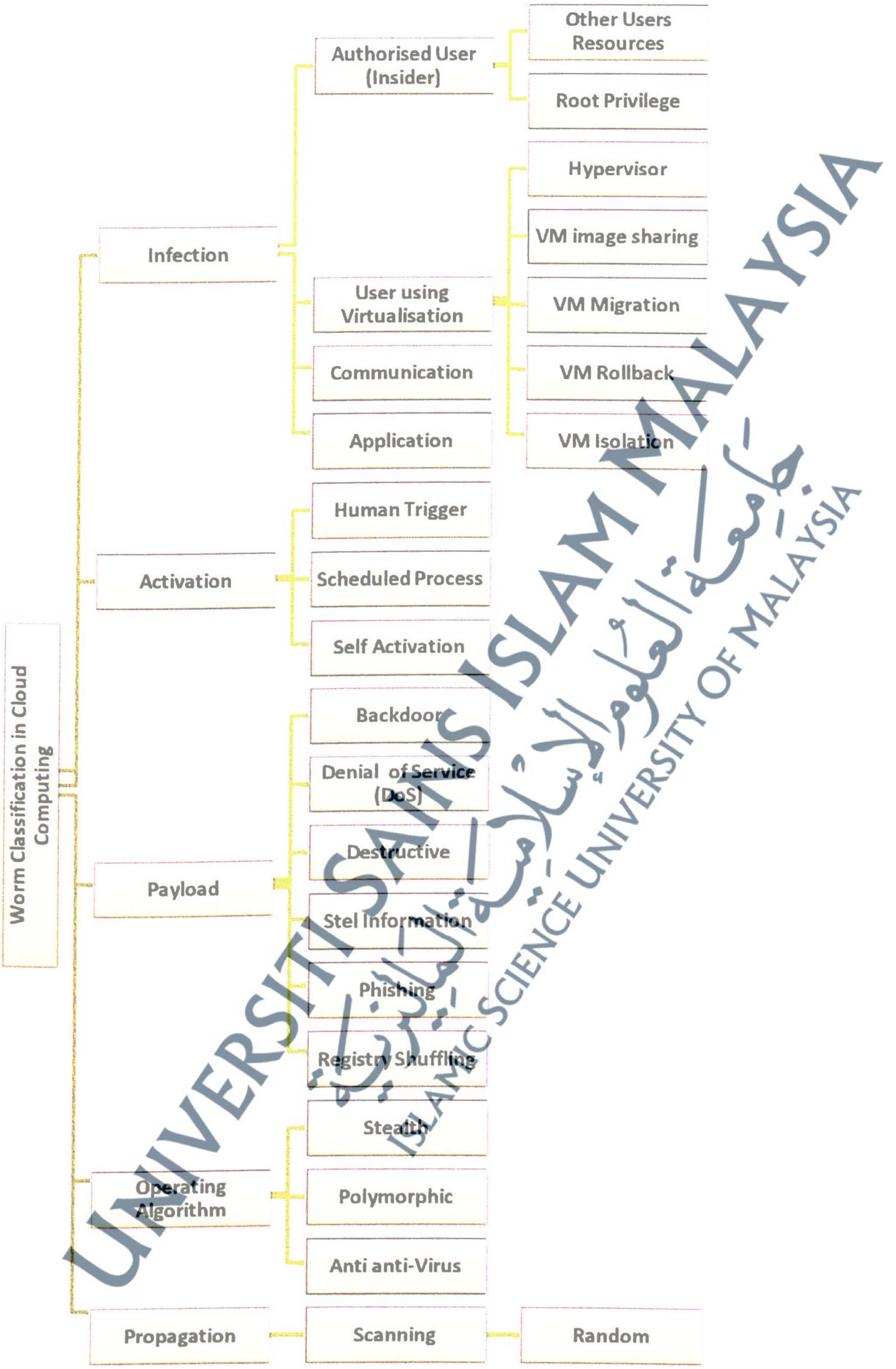

Figure 4.2: Cloud Worm Classification

In relation to the diagram above, a new cloud worm classification has been introduced. As compared with other related works on worm classification, these works have introduced different worm features that are limited to PC environment. This research proposes and implements a new classification for cloud worms. This research investigates and adds new sub-classes for infection part as a contribution to the new cloud worm classification.

A) Infection

Infection refers to how a computer becomes infected by a Worm (McGraw & Morrisett, 2000). The two ways a computer can be infected by worm are host or network. As for USB, CD and File, they are the most common hosts infecting computer worm. However, the main concern of this study is worm attack in cloud and therefore, this study takes into account worm attack through virtualisation.

In the infection via network, the victim's computer can be infected when he/she visits malicious sites and download the infected files without their knowledge, or when they received spam emails and execute the infected attachment. Additionally, chatting channels such as Internet Relay Chat (IRC) is a form of real-time Internet chat or synchronous conferencing or a communication channel that allows it to transfers data via messages. Worm can exploit this communication facility channels to spread itself into these paths. Internet Relay Chatting (IRC) channels are very much vulnerable to worm spreading as they are exploited for sending infected files or links to infected websites. When users receive the infected messages on these chatting channels and click the link, their computer becomes directly infected by the worm. Yahoo and Facebook messengers are also examples of these communication channels. Unlike worm that can infect a network host, cloud worm is also able to infect physical or virtual host via network. If network facility is available, cloud worm can infect hypervisor and virtual host because they have network connectivity and operating system similar to the host in a network. Under this classification, the tools used for observing worm patterns are: Wire Shark, process explorer, and process monitor.

Authorised User (Insider)

Authorised Cloud users may attempt to gain or misuse their unauthorised privileges. Insiders may spread worm to disclose information to others or commit frauds or intentionally modify information which poses a serious trust issue. For example, an internal DoS attack was demonstrated against the Amazon Elastic Computer Cloud (EC2) (Modi *et al.*, 2013). Network Wire Shark and processes monitor tools were used in observing worm pattern. Under the Infection sub-category related to "authorised user," cloud worm patterns displayed unusual changes such as modifying of information on a system, which leads to serious issues such as information disclosures or other fraudulent activities.

Root privilege is another type of attack for authorised user. Here, sniffing password is the main weapon for the attacker to get access to legitimate user accounts. This allows the attacker to exploit the vulnerabilities for gaining root level access to the system. As an example, Buffer overflows are used to generate root shells from a process running as root. It takes place when application program code overfills static buffer. The mechanisms used to secure the authentication process are a frequent target. There are no universal standard or security mechanisms which can be used to avoid security risks like keyloggers, phishing attacks, and weak password recovery workflows (Modi *et al.*, 2013). By gaining root access, user could initiate worm attack in a cloud environment. Using wire Shark, process explorer and process monitor, a sign of malicious activity was seen. There are snippets of codes for some programs that have been obfuscated as having encryption algorithm which also shows signs or patterns of malicious behaviour.

User using Virtualisation

Virtualisation is used in cloud computing and also runs through standard IP protocol (Modi *et al.*, 2013). Through virtualisation, cloud user can initiate worm attack in various ways. The main module of virtualisation is VMM or hypervisor which is responsible for VMs isolation and management. There are many worms reported in VMM that allow the attacker to take control of the hypervisor and bypass security restrictions. For example, vulnerabilities in Microsoft Virtual PC, Microsoft Virtual Server and Xen can be abused by attackers to gain privileged rights (Ali *et al.*, 2015).

In order to instantiate VMs, a VM image is used and users can create their own VM image or can use an image from the shared image repository. From the repository, users are allowed to download and upload images. Amazons image repository is an example of image repository (Ali *et al.*, 2015). Sharing of VM images in the image repositories is a common practice but can evolve as a serious threat if it is used in malicious manner. A malicious user can investigate the code of the image to look for probable attack point and can also upload an image that contains worm. This VM image will become a source of worm introduction in cloud computing environment and track user activity or expose user's confidential information. VM migration, VM rollback and VM isolation (Ali *et al.*, 2015) are the other worm attacking methods in cloud. The tools that are used for the observation were wire shark, Procmon, Regmon, process explorer, filemon, Newt, PortMon, and Process monitor.

Communication

Shared infrastructure and virtual network (Ali *et al.*, 2015) enables resource pooling, storage resources, sharing of computational, and sharing of network infrastructure components. The sharing of network components delivers attacker the window of cross-tenant attack. The vulnerability stems from the resource pooling characteristic of cloud computing and affects the IaaS service model of cloud. Protection and security mechanisms over the physical network are not able to monitor the traffic over virtualised network. This becomes a serious challenge as malicious activities of VMs go beyond the monitoring of security tools. Intrusion detection and prevention mechanisms usually depend on the traffic patterns and activities to judge the anomalies and detect the possibility of the attack. For the communication part, through the use of Network and port monitoring dynamic tools, its pattern shows that it might able to trace and record TCP and UDP activities. By observing and studying the sources and destination of the addresses, profiling is used to identify suspicious communication activity which shows whether the network pattern contains anomalistic worm attacks.

Application

Worm concentrates on events which occur in running cloud application. Applications that are used for accessing cloud facility could be considered as the entry-points of worm. Using various interfaces and devices, attacker can have access to service-based applications as network browsers and thin clients. On the other hand, social engineering-based cloud application could perform malicious activities like stealing money from bank account by providing fake user interface which appears legitimate in order to circumvent the user. Possible solutions for this could be the use of application certification and user education (Elish *et al.*, 2015). By monitoring the network activity through wire shark software and web-based analysis tool, it could be seen that worm also displays an unusual pattern. Also, process explorer is used not only to monitor a single malicious program, but it also monitors the behaviour of the whole machine. The Application part of the classification shows that worms are able to run their own operation.

B) Activation

Worm activation refers to the criteria that cause worm to become active and perform its disruptive task (Salmani *et al.*, 2009). Meanwhile, according to Saudi (2011), activation is defined as worm's trigger mechanism that tries to insert worm into the host. This work is based on the dynamic analysis and it showed that the worm is activated by human trigger, self-activation, and scheduled process. Like computer network worm, cloud worm is also able to activate physical or virtual host via network. If network facility is available, cloud worms can activate in the hypervisor or in the virtual host because they have network connectivity and operating system similar to that of the host in a network. In the case of Activation classification, numerous sub activities can also be discovered by using web-based analysis tool, network and process monitor dynamic tool.

Human Trigger

Human trigger is the slowest activation mechanism activated by worms when a user clicks on an email (Saudi *et al.*, 2008), such as the Melissa worm, or worms that copy infected files onto a shared folder, such as the Nimda worm. According to Smith et al. (2009), worms can be activated by user's actions which normally are not expected to

execute a worm, such as user's login scripts, or running an application, or when a CD or memory card is inserted into the computer. By using the network, web-based analysis tool and process monitor, it was discovered that some worms began to act only when the user is performing an activity. It occurs when the user clicks on an email or a link.

Scheduled Process

Scheduled process is the second fastest worm activation mechanism (Nellutla *et al.*, 2013). An example is automatic software updates, which can be used to install and run malicious software (e.g., a worm). Based on Smith *et al.*, (2009), scheduled process activated worms are activated by a legitimate process which hasn't been properly secured, such as a legitimate program which automatically updates itself from an infected web server. For this classification, the web-based analysis tool such as virustotal is used to observe the worm process. Using virustotal enables the monitoring of strange worm activities whenever the software performs a scheduled process such as automation updates.

Self Activation

Self activation worm is the fastest activation mechanism which is the most worrisome. This type of worm begins execution immediately after being transmitted to the target. These worms generally exploit the vulnerability in a running application. Accordingly, buffer overflow vulnerabilities are a common Target (Wang et *al.*, 2010). Meanwhile, Code Red I and II are examples of self-activating worms. Also, through the use of the web-based analysis tool such as virustotal, strange activity can be observed during buffer overflow occurrence.

C) Payload

The payload of a worm refers to the behaviours or actions taken by worm. A payload is a code in the worm designed to do more than spreading the worm. It might delete files on a host system. According to Saudi *et al.*, (2008), payload is defined as a destructive mechanism which is designed with malicious intention. As for this research on worm, payload is defined as a destructive mechanism, which can cause

damage to the victim's computer and cause loss of confidential information like credit card number, and password.

Four main destructive mechanisms have been identified in this research which are: backdoor installation, denial of services (DoS), destructive, and stealing confidential information. In the experiment conducted in the laboratory, one sample was taken which showed how worm attack works in cloud computing environment and its actions were observed. The result showed that the worm dropped a file in C directory as shown in Figure 4.3. The file Worm64.dll (C:Windows\System32\Worm64.dll) was used by the worm to attack cloud server. It damages the files, registry and all data that are stored in the server. Based on this payload and the testing conducted, it appears that cloud can be exploited and attacked by a worm. Cloud worm is also able to do payload to the physical or virtual host via network like network worm. Due to the network facility, cloud worm can pass payload to the hypervisor and virtual host because they have network connectivity and OS similar to the host in a network. Due to this reason, cloud worm can use vulnerable ports to initiate attack in the cloud. As for the Payload classification, the analysis tools that were used are network and process monitor for identifying the "Payload" patterns. It was seen to have a distinctive feature.

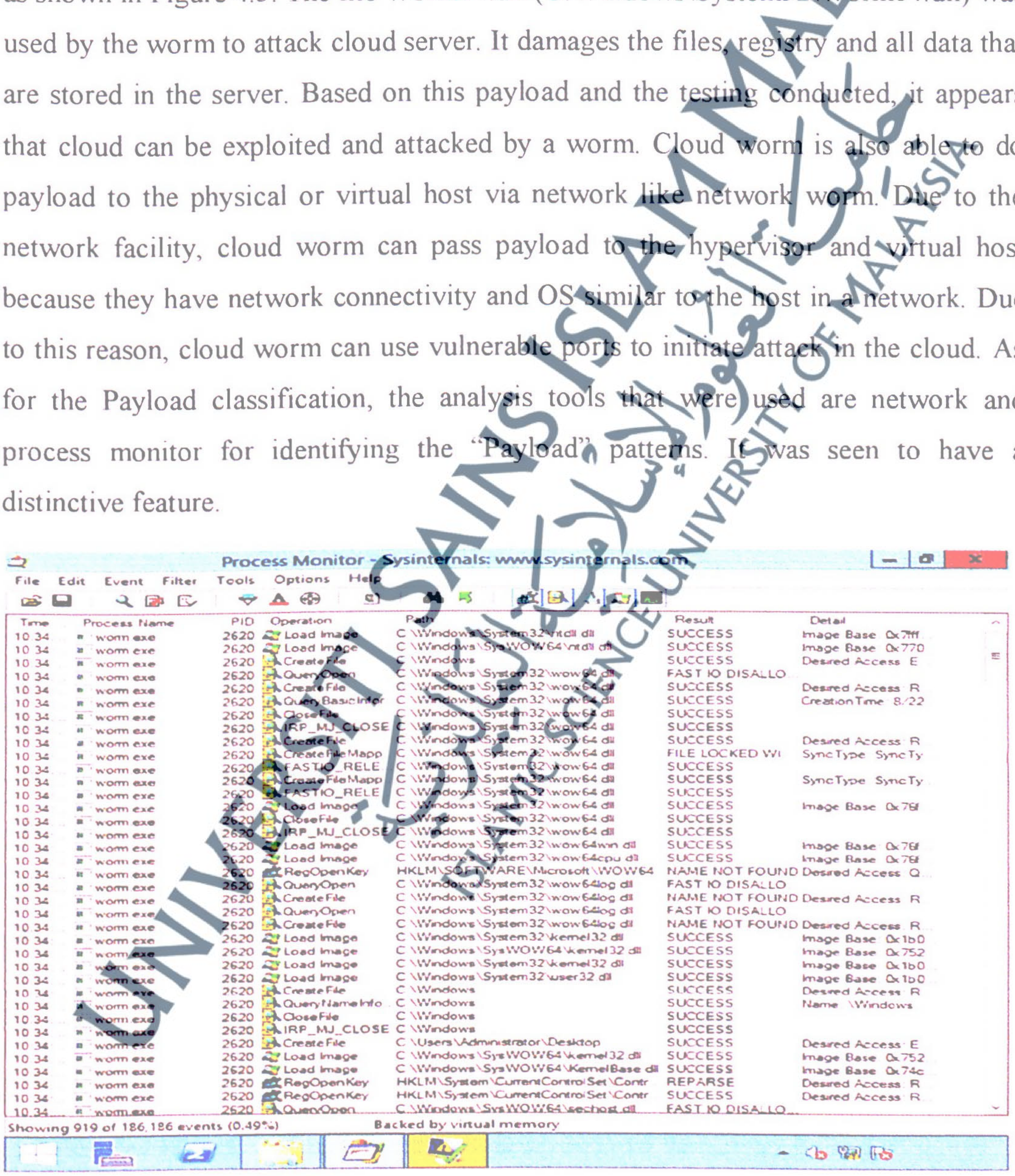

Figure 4.3: Payload worm attack in cloud for file and registry

Backdoor Installation

Installing a backdoor in the infected computer is a very common payload for worms. This allows the attacker to control victim computers. Some payloads can open backdoors on victim machines in order to make the remote control of the captured machines possible by bypassing the usual security tools (Nellutla *et al.*, 2013). In this research, installing backdoor is referred as an illegal way of entering a computer system and providing remote control via the Internet to help the attacker to control end user computers and steal confidential information. This worm pattern, through the use of wire shark, Newt monitor and process explorer, was seen as having the capability to act as a remote for the attacker to access the computers.

Denial of Service (DoS)

DoS attack is an attempt to make a computer or network resource unavailable to the users, and it has the ability to suspend or interrupt the services of a host connected to the Internet. A commonly used payload is the issuance of a denial of service attack against one or several web sites. A large worm network can cause large damage by issuing distributed dos (DDoS) attack, where all the worms simultaneously launch attacks against the same web site. Code Red worm is an example of DoS attack (Berghel, 2001). Through using wire shark and process explorer dynamic tools, some worms showed the same patterns of denial of service (DoS), which was able to prevent computers from gaining access to networks.

Destructive

Due to a pathogen, some part of the population could confront the biological death. Similarly, some server system or cloud infrastructure could be damaged by the destructive worm. Destructive worm could corrupt system permanently or delete valuable information of servers or users in cloud environment. Therefore, a super virulent worm might initiate significant challenges to its further propagation (Vlachos *et al.*, 2010). These types of attack spread through physical or virtual networks. One of the patterns discovered under this worm category using dynamic tools such as file monitor, Nessus network and process explorer, showed its activity to be destructive.

Steal Information

Worm has the capability to steal confidential information in the infected computers; it can do that by collecting username and password, personal information, email address, local IP address, and capturing credit card number. Worm targeted towards normal, everyday computers can be designed to steal credit card or bank information to perform direct money theft. Worm could be harvested through email addresses by sharing information to spammers and possible to gain remote control of the computer (LeDoux & Lakhotia, 2015). By using network and file monitor dynamic tools, another cloud worm pattern was discovered under the payload worm classification with the capability to steal information from other systems. By also using file monitor, it was observed that files that were present went missing after the worm was injected.

Phishing

The most beneficial ways for criminals to make money is by identity theft; this is also referred as phishing (Hinde, 2005). It could be found on a web server and also takes the form of an email. Users are tricked by a message which makes them give their bank account or identification information. Sometimes the process of messaging and tricking users is really decent by the criminals and it can be very hard to detect that it is a phishing message. Cyber criminals used even the bank logo and bank writing style. The new methods of these types of theft are performed using smart phone apps and different messaging service or even by sending messages to the mobile phone. This type of attacks is increasing day by day and is very hard to detect if people are not aware of it. According to Bureau of Justice Statistics (BJS), about 17.6 million people faced identity theft in 2014. Seven percent of U.S. residents over the age of 16 were victims of at least one identity theft incident in 2014 (Harrell, 2015). Meanwhile, in 2014, 85% of people took actions to avoid identity theft, such as shredding documents that contain personal information, checking credit reports, and changing passwords on financial accounts. The majority of victims aged more than 65, and more females (9.2 million) were victims of identity theft compared with males (8.3 million) in 2014 (Harrell, 2015). Another distinguished worm pattern that was observed using the Wire Shark process explorer and Process monitoring tool was its ability to take form of an email which enables it to copy personal information such as usernames and passwords, causing identity theft pattern.

Registry Shuffling

When a worm performs execution on an infected host, it may take control of the system with the highest privileges, modifying the system as needed, and continue to infect other hosts. These acts expose some anomalies on the infected computers, such as modifying or writing the registry keys and system binaries or opening network connections to transfer worm as executable form to other vulnerable hosts. For example, the "Blaster" worm changes a registry entry, downloads a file named "msblast.exe", and executes it (CERT, 2003). This was also one of the patterns of the worms that were observed by using wire shark, regmon and File monitoring tools. Part of the injected worm showed that it had the ability to alter files or open network whenever it invades a system.

D) Operating Algorithm

Based on Saudi (2011), operating algorithm refers to the technique used to avoid malicious code detection. Worm has the ability to hide codes in the system since it is capable of changing its location frequently and keep changing its names to avoid detection by anti-virus or any security tools installed in the machine like firewall. As for this worm research, its operating algorithms can be classified into stealth, polymorphic and anti anti-virus. Cloud worm is also able to run operating algorithm on the physical or virtual host via network. Because of the network connectivity, cloud worm is able to run operating algorithm in the hypervisor and virtual host for various purposes. They have network connectivity and operating system as a host in a network.

Stealth

Some of the worms have stealth capability. Therefore, it will try to keep itself unseen from the user of the victim's machine. Their activities are usually hidden to the end user and they generally do not consume a large percentage of system resources to avoid any suspicions.

Polymorphic

A polymorphic worm is a worm that tries to modify themselves using polymorphism, encryption or both to avoid detection by security tools. Polymorphic worm tries to

modify its code by rearranging the functional blocks of code (Kolesnikov & Lee, 2005). Since the code for the worm can change each time it is transmitted, the code of the worm will not match any worm signatures. Accordingly, Saudi *et al.*, (2008) defined polymorphic worm as a worm that changes all parts of their code each time they replicate to avoid the scanning software.

Anti Anti-virus

According to (Saudi, 2011), an Anti Anti-virus operating algorithm worm can damage anti-virus software by deleting or changing antivirus software settings, by attempts to disable anti-virus software and edit the definition data files. Worm may disable the anti-virus software on the host, which will enable it to cause more damage. Worm has the ability to attack antivirus on the targeted computer and causes its disability and failure of antivirus in helping users to clean out the worm entirely, making it impossible for the antivirus to figure out a perfect solution to handle it timely.

E) Propagation

Propagation is defined as the process where worms find new hosts to infect (Fan *et al.*, 2013; Zheng *et al.*, 2011). Propagation can be viewed as an essential phase of the life cycle of worms, where it sends packets to random hosts to propagate worm by using random scans methods. As for this research, worm can propagate itself by random scanning. Cloud worm can be able to propagate in the physical or virtual host via network like network worm. If network facility is available, cloud worm can propagate into the hypervisors and virtual hosts because they have network connectivity and OS similar to the host in a network.

Random Scanning

Random scanning is the method where the worm selects a target by a random IP address in the internet and infects it, and then continues the cycle by generating a new random target. Code-Red-I v2 worm is the example of a random scanning, so that each infected computer tries to infect a different list of randomly generated IP (Moore *et al.*, 2003). Through using wire shark dynamic analysis tools, one of the patterns observed was that the ability of the worm to choose the IP address of a random target online and then have that chosen target infected.

4.4 Cloud Worm Dynamic Analysis

The latest worm dataset was collected form virusshare; all types of worm that can infect via network are also able to infect cloud. The reason is because cloud in a complex type of computer network where all physical or virtual hosts are connected via network and like network host each physical or virtual host have their own OS. A detail description of worm dataset can be found in Section 3.3.1. All collected worm are analysed in controlled lab environment with some resources and analysis software (highlighted in Section 3.3.2 to 3.3.3). Worm analysis was done using dynamic analysis process and the detail of the analysis process can be found in Section 3.3.4. The outcome of the dynamic analysis is presented in Appendix A.

For dynamic analysis, each worm sample is tested on a controlled environment mode. After testing each worm, the controlled environment is restored to the uninfected state by using DeepFreeze software tool for analysing other samples. Initially, each worm sample is run on controlled cloud environment to test its characteristics where various dynamic analysis tools and software were running before infection occurred. If any unexpected activity is triggered by any of these tools, then the worm is identified as malicious and its characteristics are analysed carefully. On the other hand, if a sample is not identified as malicious, then the analysis process is ended and the worm is excluded from the dataset list. Virusshare report is also analysed to define a worm as malicious because of some dormant behaviour of some worms. After defining a worm as malicious, it is checked by all dynamic analysis tools one after another. After completing the analysis, it is checked whether all characteristics of sample worm are revealed or not. The analysis process is simply ended if all characteristics are found. Otherwise, the worm is analysed by the next phase. In this phase, the sample worm is further tested by virustotal web based analysis tool, which generates a report with the behaviour of the worm. Virustotal has the ability to analyse any suspicious file in a matter of seconds by executing samples in an isolated cloud environment. If all characteristics are found in this phase, the process will be ended. If not, the analysis process will run for the second time with different version of the analysis tools. About 99% of worm sample characteristics were defined in this way. However, the rest of the worm samples whose characteristics could not be defined in this way were

excluded from the dataset. A flowchart of dynamic cloud worm analysis is shown in Figure 4.4. Infection could be done by four different ways in cloud environment. Firstly, worm can initiate attack by authorised user, using virtualisation, communication or by application. Propagated worm through authorised user could be initiated as root user privilege or can be infected using other user resources like shared memory, file or disk access privilege. However, the most dangerous attack is if the worm can gain access as root user and initiate attack as root user. To define whether a worm attack is by other user's resources or is a root privilege, sample worm was executed in root mode and user mode. In case of infection by virtualisation; if a worm is able to initiate any of virtualisation related infections, it will be able to infect all VM related characteristics. Infection could also be initiated by communication access privilege and any other types of application access capability. The process of activation can be done by human trigger, scheduled process, and self activation. Worms which are activated by human trigger are easier to identify by the controlled environment. However, virusshare report is analysed mostly in two other cases. Payload, operating algorithm and propagation could be easily identified by the dynamic analysis tools. In worse case, virusshare report was also used to identify the characteristics of payload, operating algorithm and propagation.

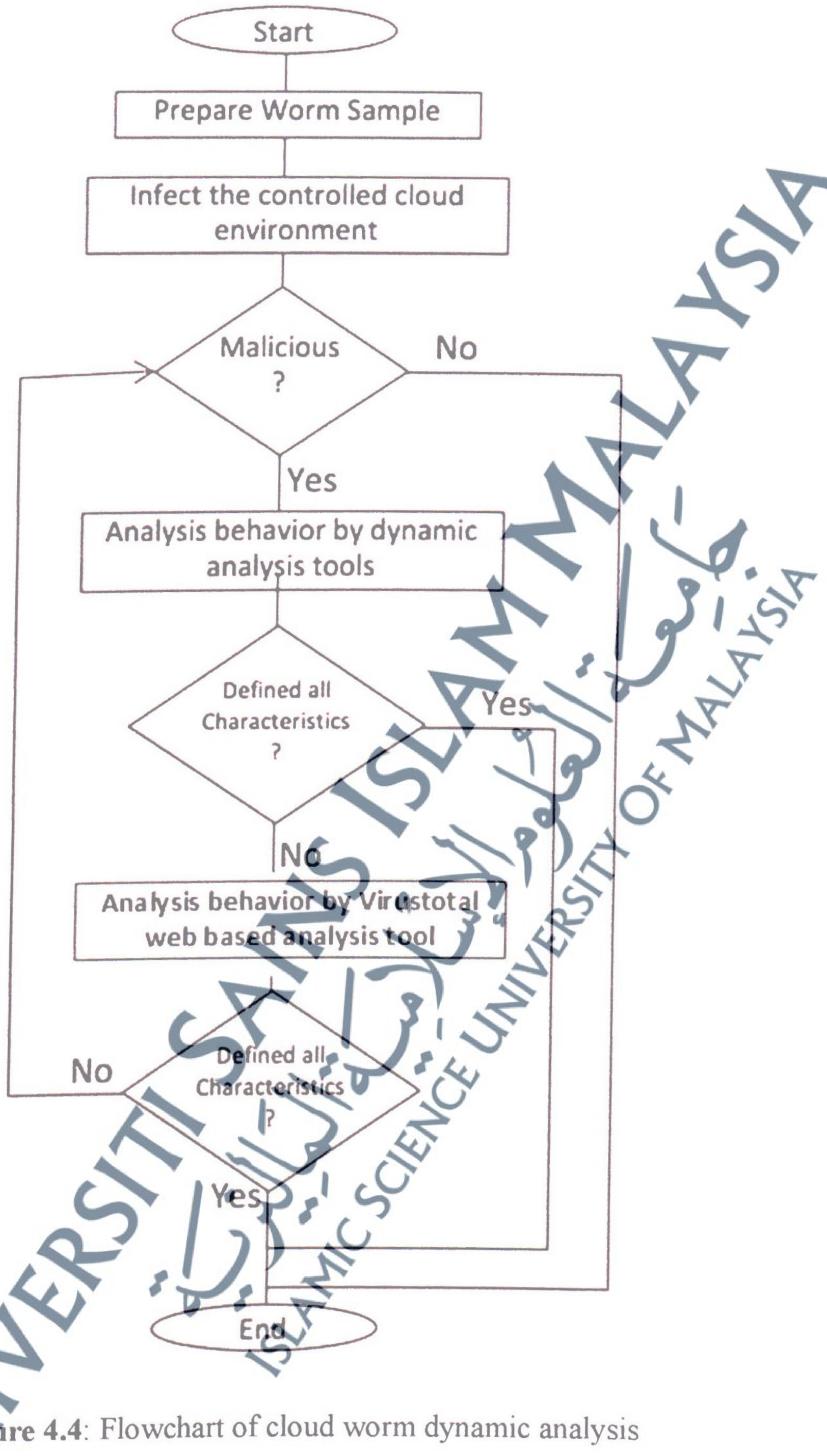

Figure 4.4: Flowchart of cloud worm dynamic analysis

4.5 Experimental Results Analysis of Worm Classification

After dynamic analysis was performed, independent test was carried out in order to find the relationship among worm characteristics. Chi-square and symmetric measure testing was initiated on the obtained dataset. The detail description of Chi-square and symmetric measures can be found in section 3.3.6.5. Meanwhile, a comprehensive description of data mining is presented in section 3.3.6.7.

4.5.1 Categorised Frequency Analysis

Frequency analysis helps to identify the most important worm detection attributes, and also helps to determine the relationship between the attributes. Frequency analysis of infection shows that most of the attack initiated through application is 14%. Through hypervisor, VM image sharing, VM migration, VM rollback, VM isolation, and communication attack could be initiated by 11.9% for each. A Pie chart in Figure 4.5 shows nine different infection types of worm attacks in cloud. The lowest number of worm attack initiated by root privilege is by 5.8%, followed by other users' resources (8.6%). According to Figure 4.5, about 60% of the attacks are initiated through virtualisation and VM related services which are the main tools or technology for cloud deployment. This is the main reason to develop a new model for worm detection and response in cloud.

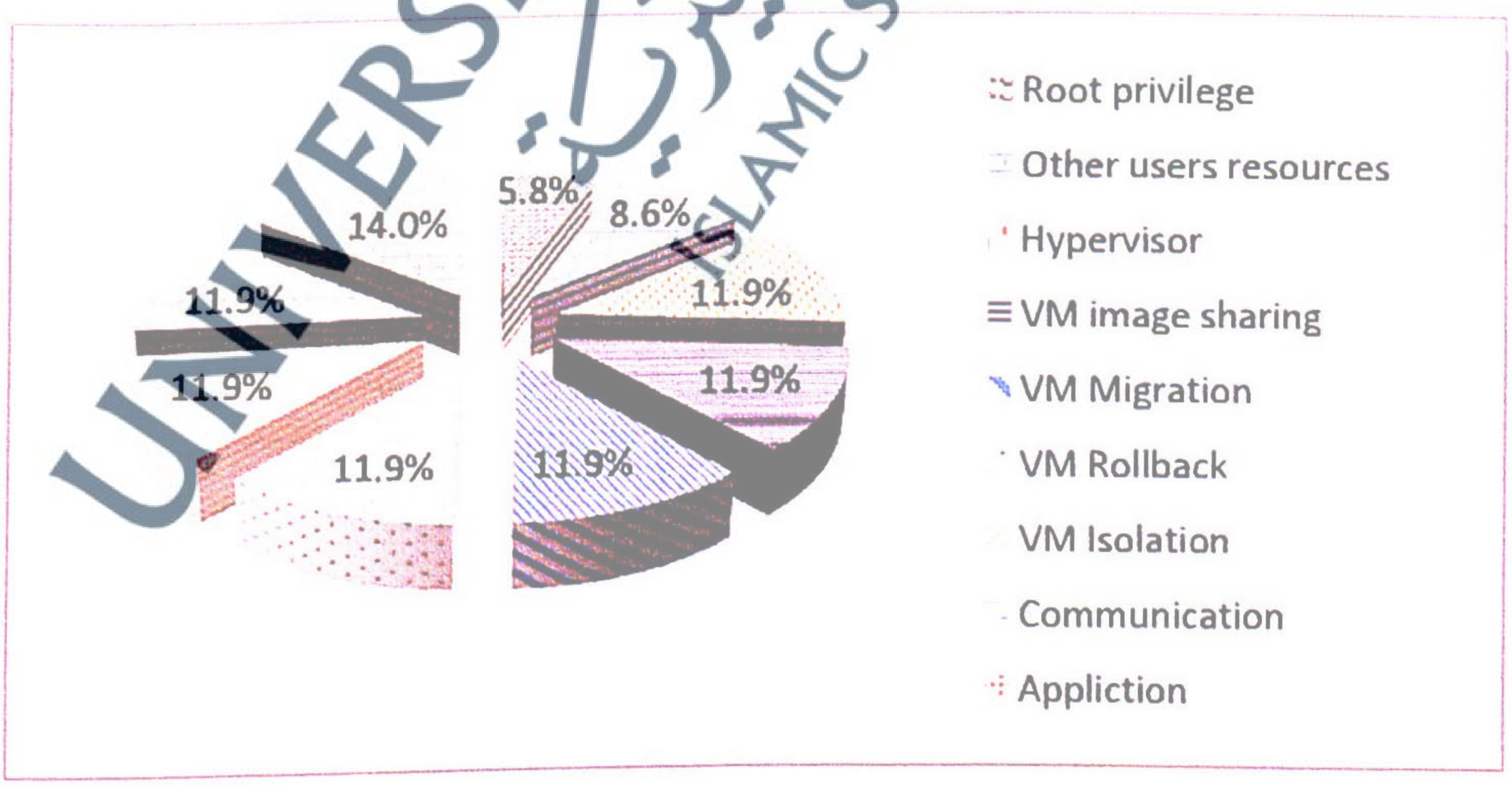

Figure 4.5: Analysis of infection result

According to the activation analysis result, 41.8% of worm attacks are activated through human trigger. An analysis of activation result shown in Figure 4.6 shows the lowest number of activation commenced by scheduled process which is at 19.7%. However, self-activation is carried out by 38.5% considering collected dataset.

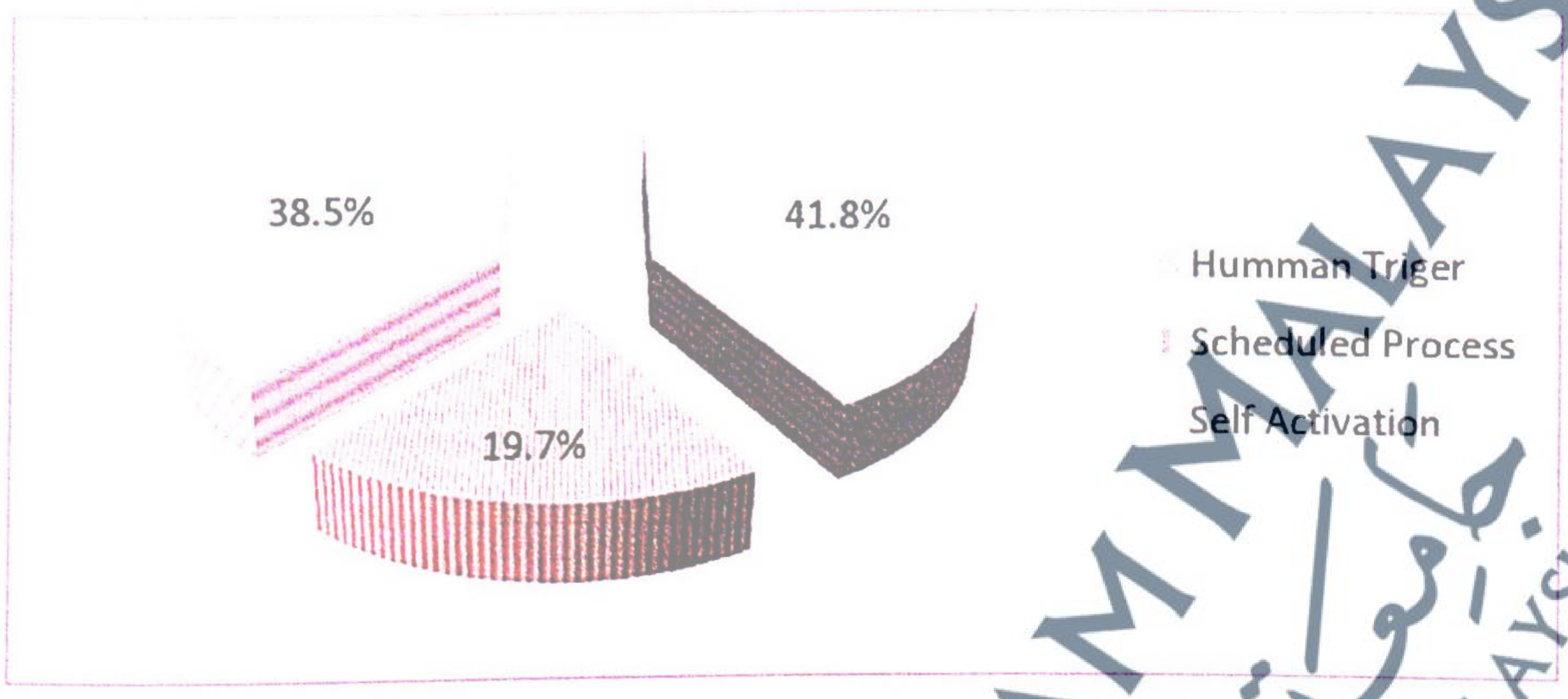

Figure 4.6: Analysis of activation result

Following the payload analysis results, only 12% worm payload is destruction which is the lowest payload method compared with other payload methods. The maximum number of payload initiated by using registry shuffling is 20.7%. Backdoor, DoS, steal information and phishing payload is 13.3%, 18.8%, 19% and 16.4% respectively. The result analysis of payload data is presented in Figure 4.7.

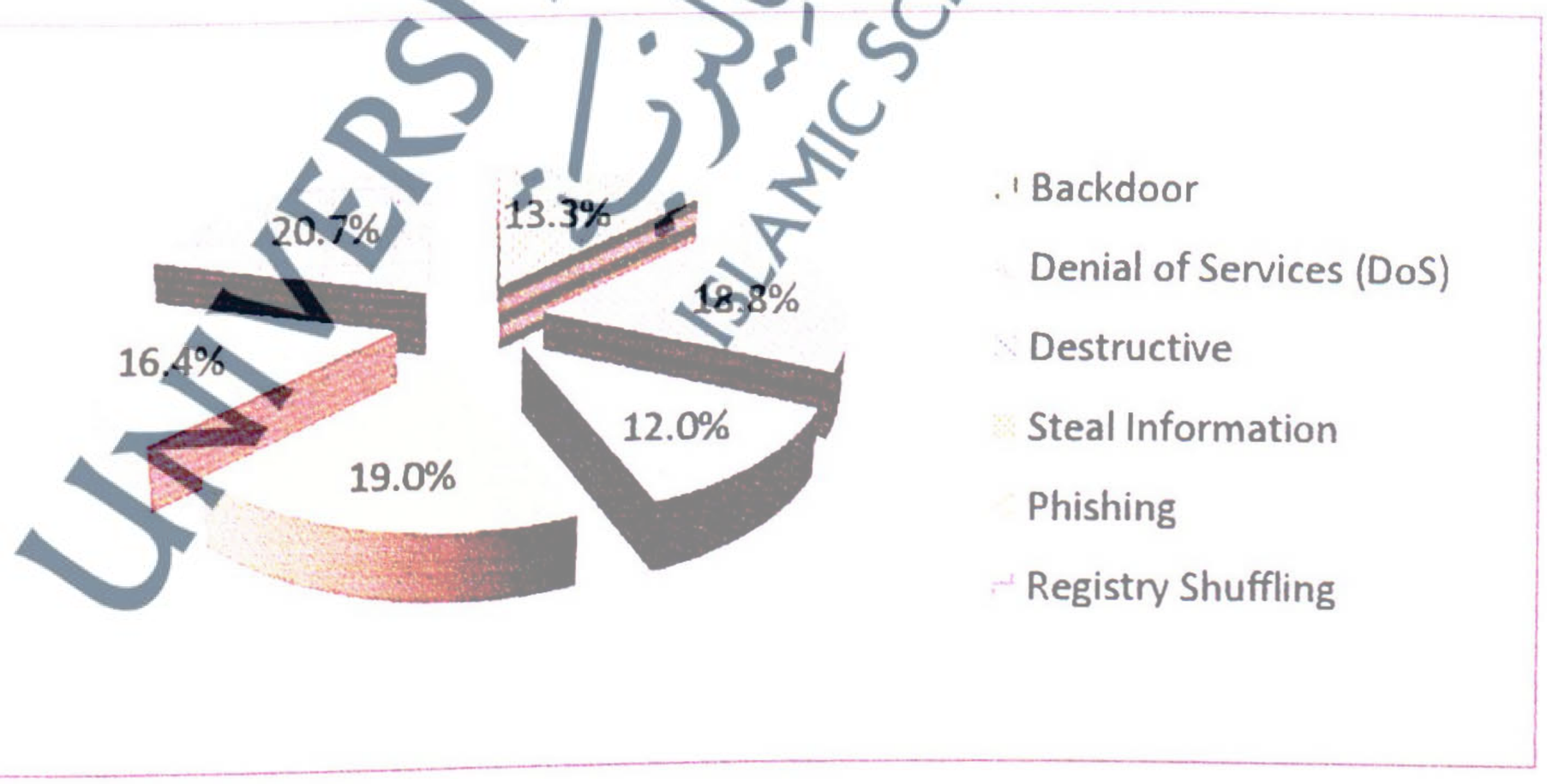

Figure 4.7: Analysis of payload result

About 43.9% of worm operating algorithm is stealth; this attribute is a big threat for cloud users because of its personal account and bank account information stealing behaviour. Anti anti-virus is another dominant operating algorithm representing 36.3% of attacks. On the other hand, polymorphic operating algorithm is less dominant by 19.8% from total dataset. An analysis result of operating algorithm is shown in Figure 4.8.

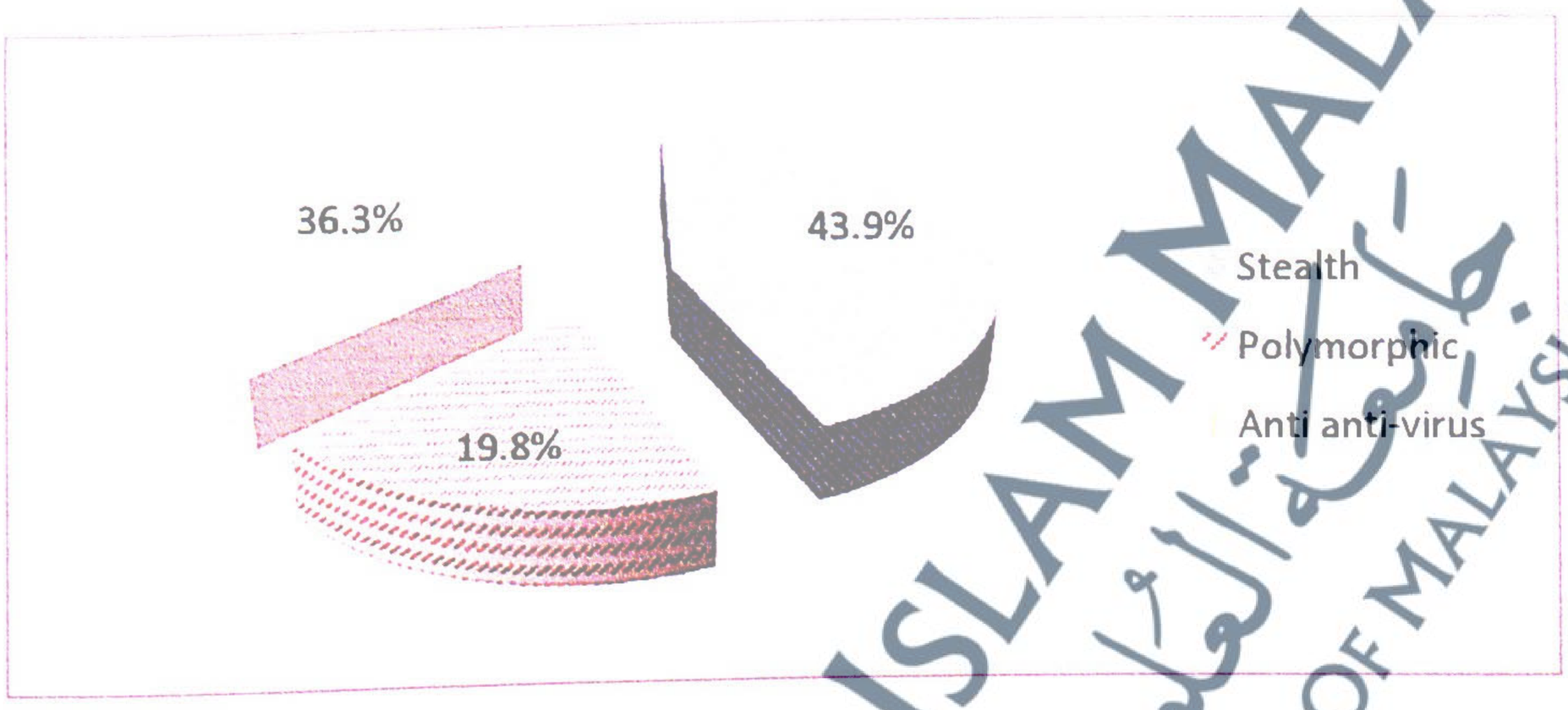

Figure 4.8: Analysis of operating algorithm result

Figure 4.9 illustrates the analysis results of propagation. As can be observed, about 65.6% worm propagation is random and the rest of worm propagation (34.4%) is not random.

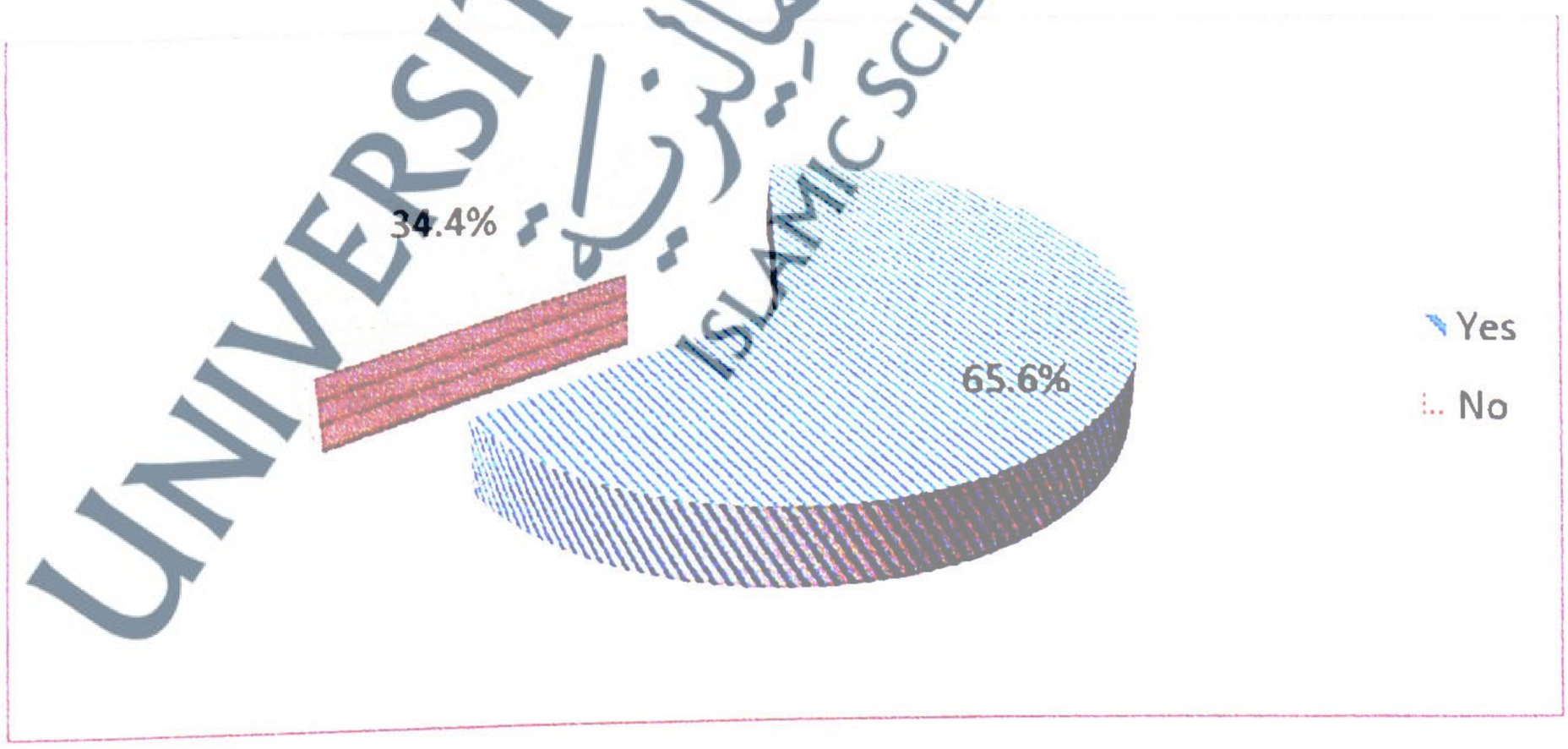

Figure 4.9: Analysis of propagation result

From the infection, it could be summarised that most of infections are related to hypervisor and VM. In activation, self activation and human trigger mostly happen. In payload, only DoS causes service interruption, but other methods could be damaging cloud seriously. Operating algorithm could be destructive when it's related to stealth. However, other methods are also damaging to cloud services, but partially. Finally, maximum worm has the behaviour of random propagation which could damage cloud heavily.

4.5.2 Statistical Analysis Result Exploration

In this study, the experiments were conducted using samples collected from virusshare using the controlled laboratory environment as stated in section 3.3.2. The methodology used for this test is described in Chapter 3 of this thesis. The objective of statistical analysis is to find out the relationship between various sub-features of infection, activation and payload cloud worm classification. Chi-square and symmetric measured statistical tests were carried out and the test results are presented in this section. A total number of 57 tests were conducted for chi-square test. The reason for 57 tests is that, there are five main features of cloud worm classification (Infection, Activation, Payload, Operating Algorithm and propagation). Every feature has one or more sub-features except propagation. Propagation has only one sub feature that is random scanning. Due to this, Chi-square test is not possible for propagation's feature. However, the rest of other four features namely infection, activation, payload, and operating algorithm, has nine, three, six and three sub-features respectively. So, if it is needed to run chi-square test with each other in every feature, then, there would be 57 chi-square tests. Infection has 9 sub-features and if there is a need to combine two features at a time; then, the total combination will be 36 as shown in Equation (4.1).

$$^{9}C_{2} = \frac{9\times8}{2!} = 36 \qquad (4.1)$$

Similarly, activation has 3 sub-features, and if there is a need to combine two features at a time; then, the total combination will be 3 as shown in Equation (4.2).

$$^{3}C_{2} = \frac{3\times2}{2!} = 3 \qquad (4.2)$$

Propagation has 6 sub-features, and if there is a need to combine two features at a time; then, the total combination will be 15 as shown in Equation (4.3).

$$^{6}C_{2} = \frac{6\times5}{2!} = \mathbf{15} \qquad \textbf{(4.3)}$$

On the other hand, operating algorithm has 3 sub-features and if there is a need to combine two features at a time; then, the total combination will be 3 as shown in Equation (4.4).

$$^{3}C_{2} = \frac{3\times2}{2!} = \mathbf{3} \qquad \textbf{(4.4)}$$

Finally, a total of 57 chi-square tests have to be run in order to find out the relationship between various sub-features. From each group, one test result is presented in the following. All results are incorporated in appendix B of this thesis. The obtained results are summarised and described briefly at the end of this section.

The relationship between every feature that was tested was found out using Chi-square test using SPSS software as explained in chapter 3, section 3.3.6.5. Figure 4.10 shows the sample of the data input into SPSS software for the performance of the Chi-square test in order to find the relationship between various sub-features.

In a research that was carried out by Saudi (2011), the Chi-square was also implemented to perform Symmetric Measures in order to perform the data mining related to the study on STAKCERT. It was also used to find out the relationship existing between worm characteristics that were chosen for the STACKCERT relation model.

In another research by Sadiqqui et al. (2009), Chi-square was used to carry out test of independence in order for them to determine whether relationship exists between the feature and target variable. They used Chi-square to help them in their study to detect internet worms during their data mining techniques.

*Version.sav [DataSet1] - IBM SPSS Statistics Data Editor

File Edit View Data Transform Analyze Direct Marketing Graphs Utilities Add-ons Window Help

Visible: 22 of 22 Variables

	Rootprivlege	Otherusersresources	Hypervisor	VMimagesharing	VMmigration	VMRollback	VMisolation	Communication	Appliction	HummanTriger	ScheduledProcess	SelfActivation	Backdoor	DenialofService
1	2	1	2	2	2	2	2	2	1	1	2	1	1	2
2	2	1	1	1	1	1	1	1	2	1	1	1	1	1
3	1	2	1	1	1	1	1	1	1	1	2	1	1	2
4	1	2	1	1	1	1	1	1	2	2	1	1	2	1
5	1	2	1	1	1	1	1	1	2	2	2	1	1	2
6	2	1	1	1	1	1	1	1	1	2	1	1	1	2
7	1	2	1	1	1	1	1	1	2	1	2	2	1	2
8	1	2	1	1	1	1	1	1	1	1	1	1	1	1
9	1	2	2	2	2	2	2	2	1	2	1	1	2	1
10	2	1	2	2	2	2	2	2	1	1	2	1	2	2
11	1	2	1	1	1	1	1	1	2	1	2	2	1	1
12	2	1	2	2	2	2	2	2	1	1	2	2	1	1
13	2	1	2	2	2	2	2	2	2	2	1	1	1	2
14	1	2	1	1	1	1	1	1	1	1	2	1	1	1
15	2	1	2	2	2	2	2	2	1	1	1	2	1	1
16	1	2	1	1	1	1	1	1	1	2	1	1	1	1
17	1	2	2	2	2	2	2	2	1	1	2	1	1	1
18	1	2	2	2	2	2	2	2	1	1	1	1	1	1
19	1	2	1	1	1	1	1	1	1	1	2	2	1	1
20	2	1	1	1	1	1	1	1	2	1	2	2	1	2
21	1	2	2	2	2	2	2	2	1	2	2	1	2	1
22	1	2	2	2	2	2	2	2	1	1	1	2	1	1

Data View Variable View

IBM SPSS Statistics Processor is ready

File Edit View Data Transform Analyze Direct Marketing Graphs Utilities Add-ons Window Help

	Destructive	StealInformation	Phishing	RegistryShuffling	Stealth	Polymorphic	AntiantiVirus	ScaninningRandom
1	1	2	2	1	1	1	2	1
2	1	1	2	1	1	2	1	1
3	1	1	2	1	1	2	2	1
4	1	1	2	1	1	2	2	1
5	2	1	2	1	1	1	2	1
6	1	1	2	1	1	1	2	1
7	2	1	2	1	1	2	2	1
8	1	1	1	1	1	1	1	1
9	2	1	1	1	1	2	2	2
10	1	1	1	1	1	1	2	2
11	1	1	2	1	1	2	1	1
12	2	2	1	1	1	1	1	1
13	2	1	1	1	1	2	1	1
14	2	1	2	1	2	2	1	1
15	1	1	1	1	1	1	1	1
16	1	1	1	1	2	2	2	1
17	1	1	1	1	1	1	2	1
18	2	2	1	1	1	1	2	1
19	2	1	2	1	2	2	1	1
20	2	2	2	1	1	2	2	1
21	1	2	2	1	1	2	1	2
22	2	2	2	1	1	2	1	1

Data View Variable View

Figure 4.10: Sample data for Chi- square test

4.5.2.1 Results for the relationship between root privilege and other users resources

Table 4.1 shows the results for the relationship between root privilege and other users' resources.

Table 4.1: The relationship between root privilege and other users' resources

Root Privilege * Other Users Resources Crosstabulation					
			Otherusersresources		Total
			Yes	No	
Root Privilege	Yes	Count	0	411	411
		Expected Count	245.1	165.9	411.0
		% within Rootprivilege	0.0%	100.0%	100.0%
		% within Otherusersresources	0.0%	100.0%	40.4%
		% of Total	0.0%	40.4%	40.4%
	No	Count	607	0	607
		Expected Count	361.9	245.1	607.0
		% within Rootprivilege	100.0%	0.0%	100.0%
		% within Otherusersresources	100.0%	0.0%	59.6%
		% of Total	59.6%	0.0%	59.6%
Total		Count	607	411	1018
		Expected Count	607.0	411.0	1018.0
		% within Rootprivilege	59.6%	40.4%	100.0%
		% within Otherusersresources	100.0%	100.0%	100.0%
		% of Total	59.6%	40.4%	100.0%

Chi-Square Tests					
	Value	Df	Asymp. Sig. (2-sided)	Exact Sig. (2-sided)	Exact Sig. (1-sided)
Pearson Chi-Square	1018.000[a]	1	.000	.000	.000
Continuity Correction[b]	1013.850	1	.000	0.000	.000
Likelihood Ratio	1373.274	1	.000	0.000	.000
Fisher's Exact Test	.000	0	.000	.000	.000
Linear-by-Linear Association	1017.000	1	.000	.000	.000
N of Valid Cases	1018	0	.000	.000	.000

a. 0 cells (0.0%) have expected count less than 5. The minimum expected count is 165.93.

b. Computed only for a 2x2 table

Symmetric Measures			
		Value	Approx. Sig.
Nominal by Nominal	Phi	-1.000-	.000
	Cramer's V	1.000	.000
N of Valid Cases		1018	.000

It was assumed that if the null hypothesis (H_0) is accepted, then there would be no relationship between root privilege and other users' resources. On the contrary, if the alternate hypothesis (H_a) is accepted, then a relationship between root privilege and other users' resources is present. This is the standard hypothesis testing in various research fields (Leedy & Ormrod, 2005).

According to the statistical analysis results in Table 4.1 above, the relationship between root privilege and other users' resources has a strong negative relationship with Pearson Chi-Square value which is 1018.000, probability (p) or significance value of 0.00 and Phi value is -1.000 using the Chi-Square and symmetric measure tests. So, Ha is accepted and H0 is rejected, while p value is less than 0.05. These results indicate that the relationship did not happen by chance and it is based on Chi-Square tests. The value of probability (p) for the distribution occurring by chance is 0.00 as presented in Table 4.1. In conclusion, it is proved that there is a relationship between root privilege and other users' resources.

4.5.2.2 Results for the relationship between human trigger and scheduled process

Table 4.2 shows the results for the relationship between human trigger and scheduled process.

Table 4.2: Statistical results of the relationship between human trigger and scheduled process

Human Trigger * Scheduled Process Crosstabulation					
			ScheduledProcess		Total
			Yes	No	
Human Trigger	Yes	Count	456	539	995
		Expected Count	459.4	535.6	995.0
		% within HumanTrigger	45.8%	54.2%	100.0%
		% within ScheduledProcess	97.0%	98.4%	97.7%
	No	Count	14	9	23
		Expected Count	10.6	12.4	23.0
		% within HumanTrigger	60.9%	39.1%	100.0%
		% within ScheduledProcess	3.0%	1.6%	2.3%
Total		Count	470	548	1018
		Expected Count	470.0	548.0	1018.0
		% within HumanTrigger	46.2%	53.8%	100.0%
		% within ScheduledProcess	100.0%	100.0%	100.0%

Chi-Square Tests					
	Value	df	Asymp. Sig. (2-sided)	Exact Sig. (2-sided)	Exact Sig. (1-sided)
Pearson Chi-Square	2.046[a]	1	.153	.000	.000
Continuity Correction[b]	1.486	1	.223	.000	.000
Likelihood Ratio	2.045	1	.153	.000	.000
Fisher's Exact Test	0	0	.000	.204	.112
Linear-by-Linear Association	2.044	1	.153	.000	.000
N of Valid Cases	1018	0	.000	.000	.000

a. 0 cells (0.0%) have expected count less than 5. The minimum expected count is 10.62.

b. Computed only for a 2x2 table

Symmetric Measures			
		Value	Approx. Sig.
Nominal by Nominal	Phi	-.045-	.153
	Cramer's V	.045	.153
N of Valid Cases		1018	.000

From to the statistical analysis results presented in Table 4.2 above, the relationship between human trigger and scheduled process has a weak negative relationship with Pearson Chi-Square value being 2.046, probability (p) or significance value of 0.153 and Phi value of -0.045 using Chi-Square and symmetric measure tests. So, H_0 is accepted and H_a is rejected, while p value is greater than 0.05. These results indicate that the relationship did occur by chance and it is based on Chi-Square tests. The value of probability (p) for the distribution occurring by chance is 0.153 as presented in Table 4.2. In conclusion, it is proved that there is no relationship between human trigger and scheduled process.

4.5.2.3 Results for the relationship between backdoor and Denial of Services (DoS)

Table 4.3 shows the results for the relationship between backdoor and Denial of Services (DoS).

Table 4.3: Statistical results of the relationship between backdoor and Denial of Services (DoS)

Backdoor * Denial of Services Crosstabulation				
			DenialofServices	Total
			Yes	No	
Backdoor	Yes	Count	604	47	651
		Expected Count	589.6	61.4	651.0
		% within Backdoor	92.8%	7.2%	100.0%
		% within DenialofServices	65.5%	49.0%	63.9%
		% of Total	59.3%	4.6%	63.9%
	No	Count	318	49	367
		Expected Count	332.4	34.6	367.0
		% within Backdoor	86.6%	13.4%	100.0%
		% within DenialofServices	34.5%	51.0%	36.1%
		% of Total	31.2%	4.8%	36.1%
Total		Count	922	96	1018
		Expected Count	922.0	96.0	1018.0

	% within Backdoor	90.6%	9.4%	100.0%
	% within DenialofServices	100.0%	100.0%	100.0%
	% of Total	90.6%	9.4%	100.0%

Chi-Square Tests

	Value	df	Asymp. Sig. (2-sided)	Exact Sig. (2-sided)	Exact Sig. (1-sided)
Pearson Chi-Square	10.332[a]	1	.001	.000	.000
Continuity Correction[b]	9.626	1	.002	.000	.000
Likelihood Ratio	9.947	1	.002	.000	.000
Fisher's Exact Test	.000	0	.000	.002	.001
Linear-by-Linear Association	10.322	1	.001	.000	.000
N of Valid Cases	1018		.000	.000	.000

a. 0 cells (0.0%) have expected count less than 5. The minimum expected count is 34.61.

b. Computed only for a 2x2 table

Symmetric Measures

		Value	Approx. Sig.
Nominal by Nominal	Phi	.101	.001
	Cramer's V	.101	.001
N of Valid Cases		1018	.000

According to the statistical analysis results shown in Table 4.3 above, the relationship between backdoor and Denial of Services (DoS) is weak and positive with Pearson Chi-Square value at 10.332, probability (p) or significance value at 0.001 and Phi value at 0.101 using the Chi-Square and symmetric measure tests. So, H_a is accepted and H_0 is rejected while p value is less than 0.05. These results indicate that the relationship did not happen by chance and it is based on Chi-Square tests. The value of probability (p) for the distribution occurring by chance is 0.001 as presented in Table 4.3. In conclusion, it is proved that there is a relationship between backdoor and Denial of Services (DoS).

4.5.2.4 Results for the relationship between stealth and polymorphic

Table 4.4 shows the results for the relationship between Stealth and Polymorphic.

Table 4.4: Statistical results of the relationship between backdoor and Denial of Services (DoS)

Stealth * Polymorphic Crosstabulation					
			Polymorphic		Total
			Yes	No	
Stealth	Yes	Count	440	545	985
		Expected Count	430.6	554.4	985.0
		% within Stealth	44.7%	55.3%	100.0%
		% within Polymorphic	98.9%	95.1%	96.8%
		% of Total	43.2%	53.5%	96.8%
	No	Count	5	28	33
		Expected Count	14.4	18.6	33.0
		% within Stealth	15.2%	84.8%	100.0%
		% within Polymorphic	1.1%	4.9%	3.2%
		% of Total	0.5%	2.8%	3.2%
Total		Count	445	573	1018
		Expected Count	445.0	573.0	1018.0
		% within Stealth	43.7%	56.3%	100.0%
		% within Polymorphic	100.0%	100.0%	100.0%
		% of Total	43.7%	56.3%	100.0%

Chi-Square Tests					
	Value	df	Asymp. Sig. (2-sided)	Exact Sig. (2-sided)	Exact Sig. (1-sided)
Pearson Chi-Square	11.308[a]	1	.001	.000	.000
Continuity Correction[b]	10.140	1	.001	.000	.000
Likelihood Ratio	12.753	1	.000	.000	.000
Fisher's Exact Test				.001	.000
Linear-by-Linear Association	11.297	1	.001	.000	.000
N of Valid Cases	1018	0	.000	.000	.000

a. 0 cells (0.0%) have expected count less than 5. The minimum expected count is 14.43.

b . Computed only for a 2x2 table

Symmetric Measures			
		Value	Approx. Sig.
Nominal by Nominal	Phi	.105	.001
	Cramer's V	.105	.001
N of Valid Cases		1018	.000

According to the statistical analysis results shown in Table 4.4 above, the relationship between stealth and polymorphic is weak and positive with Pearson Chi-Square value being 11.308, probability (p) or significance value at 0.001 and Phi value at 0.105 using Chi-Square and symmetric measure tests. So, Ha is accepted and H0 is rejected while p value is less than 0.05. These results indicate that the relationship did not happen by chance and it is based on Chi-Square tests. The value of probability (p) for the distribution occurring by chance is 0.001 as presented in Table 4.4. In conclusion, it is proved that there is a relationship between stealth and polymorphic. Table 4.5 shows Chi-square test and symmetric measure test result for Infection.

Table 4.5: Chi-square test and symmetric measure test result for Infection

Finding s No.	Relationship between		Chi-Square Test		Symmetric Measure	Outcome
			Pearson Chi-Square value	Probability (p) or significance value	Phi value	
1	Root Privilege	Other Users Resources	1018	0.000	-1.000	There is a relationship
2	Root Privilege	Hypervisor	80.329	0.000	0.281	There is a relationship
3	Root Privilege	VM image sharing	80.329	0.000	0.281	There is a relationship
4	Root Privilege	VM Migration	80.329	0.000	0.281	There is a relationship
5	Root Privilege	VM Rollback	82.057	0.000	0.284	There is a relationship
6	Root Privilege	VM Isolation	80.329	0.000	0.281	There is a relationship
7	Root Privilege	Communication	80.329	0.000	0.281	There is a relationship
8	Root Privilege	Application	1.518	0.218	-0.039	There is no relationship
9	Other Users Resources	Hypervisor	80.329	0.000	-0.281	There is a relationship
10	Other Users Resources	VM image sharing	80.329	0.000	-0.281	There is a relationship
11	Other Users Resources	VM Migration	80.329	0.000	-0.281	There is a relationship

12	Other Users Resources	VM Rollback	82.057	0.000	-0.284	There is a relationship
13	Other Users Resources	VM Isolation	80.329	0.000	-0.281	There is a relationship
14	Other Users Resources	Communication	80.329	0.000	-0.281	There is a relationship
15	Other Users Resources	Application	1.518	0.218	0.039	There is no relationship
16	Hypervisor	VM image sharing	1018	0.000	1.000	There is a relationship
17	Hypervisor	VM Migration	1018	0.000	1.000	There is a relationship
18	Hypervisor	VM Rollback	1004.171	0.000	0.993	There is a relationship
19	Hypervisor	VM Isolation	1018	0.000	1.000	There is a relationship
20	Hypervisor	Communication	1018	0.000	1.000	There is a relationship
21	Hypervisor	Application	0.740	0.390	-0.027	There is no relationship
22	VM image sharing	VM Migration	1018	0.000	1.000	There is a relationship
23	VM image sharing	VM Rollback	1004.171	0.000	0.993	There is a relationship
24	VM image sharing	VM Isolation	1018	0.000	1.000	There is a relationship
25	VM image sharing	Communication	1018	0.000	1.000	There is a relationship
26	VM image sharing	Application	0.740	0.390	-0.027	There is no relationship
27	VM Migration	VM Rollback	1004.171	0.000	0.993	There is a relationship
28	VM Migration	VM Isolation	1018	0.000	1.000	There is a relationship
29	VM Migration	Communication	1018	0.000	1.000	There is a relationship
30	VM Migration	Application	0.740	0.390	-0.027	There is no relationship
31	VM Rollback	VM Isolation	1004.171	0.000	0.993	There is a relationship
32	VM Rollback	Communication	1004.171	0.000	0.993	There is a relationship
33	VM Rollback	Application	0.781	0.378	-0.028	There is no relationship
34	VM Isolation	Communication	1018	0.000	1.000	There is a relationship
35	VM Isolation	Application	0.740	0.390	-0.027	There is no relationship
36	Communication	Application	0.740	0.390	-0.027	There is no relationship

From the analysis of relationship among infection sub-features as shown in Table 4.5, it was found that application sub-feature has no relationship with other sub features. Except application, all other sub features has a relationship with each other. Table 4.6 shows Chi-square test and symmetric measure test result for Activation.

Table 4.6: Chi-square test and symmetric measure test result for Activation

Findings No.	Relationship between		Chi-Square Test		Symmetric Measure	Outcome
			Pearson Chi-Square value	Probability (p) or significance value	Phi value	
37	Human Trigger	Scheduled Process	2.046	0.153	-0.045	There is no relationship
38	Human Trigger	Self Activation	0.796	0.372	-0.028	There is no relationship
39	Scheduled Process	Self Activation	28.265	0.000	0.167	There is a relationship

From the chi-square test and symmetric measure test on activation as shown in Table 4.6, it was found that there is no relationship among human trigger, scheduled process and self activation. However, it was also found that there is a relationship between scheduled process and self activation because scheduled process does not need human involvement. Table 4.7 shows Chi-square test and symmetric measure test result for Payload.

Table 4.7: Chi-square test and symmetric measure test result for Payload

Findings No.	Relationship between		Chi-Square Test		Symmetric Measure	Outcome
			Pearson Chi-Square value	Probability (p) or significance value	Phi value	
40	Backdoor	Denial of Services (DoS)	10.332	0.001	0.101	There is a relationship
41	Backdoor	Destructive	198.407	0.000	0.441	There is a relationship
42	Backdoor	Steal Information	97.163	0.000	0.309	There is a relationship
43	Backdoor	Phishing	37.93	0.000	-0.193	There is a relationship
44	Backdoor	Registry Shuffling	1.696	0.193	-0.041	There is no relationship
45	Denial of Services (DoS)	Destructive	2.549	0.110	0.05	There is no relationship
46	Denial of Services (DoS)	Steal Information	3.883	0.049	-0.062	There is no relationship

47	Denial of Services (DoS)	Phishing	11.179	0.001	0.105	There is a relationship
48	Denial of Services (DoS)	Registry Shuffling	11.541	0.001	0.106	There is a relationship
49	Destructive	Steal Information	62.244	0.000	0.247	There is a relationship
50	Destructive	Phishing	1.175	0.278	0.034	There is no relationship
51	Destructive	Registry Shuffling	4.098	0.043	0.063	There is no relationship
52	Steal Information	Phishing	21.612	0.000	0.146	There is a relationship
53	Steal Information	Registry Shuffling	0.278	0.598	-0.017	There is no relationship
54	Phishing	Registry Shuffling	11.238	0.001	0.105	There is a relationship

From chi-square test and symmetric measure test result for Payload as shown in Table 4.7 above, some sub-features have relation and some do not. This is because these are related to the nature of attack and its destructiveness. For example, the relationship between DoS and destructive appears to be negative based on the significant value obtained (0.110) because DoS only causes service interruption, and so, it is not destructive. According to Vermaat et al. (2015), DoS is an assault with the purpose of disrupting any computer access to and from an internet service which could either be to a website or an email. Additionally, DoS attacks have been able to stop computer operations temporarily. With all these DoS activities, they still do not corrupt or destroy files within a computer system or services. On the other hand, stealth information could be destructive so there is a relationship between these two sub features.

In the relationship between DoS and Phishing, first of all, the reason for obtaining the relationship between DoS and phishing is due to the analysis carried out using the Chi-square through SPSS software. Secondly, according to Zeltser (2016), DoS can be utilised for distraction in order to do other malicious actions. The phisher may carry out a DoS attack in order to draw the attention of authorities away from other fraudulent activities which might be performed elsewhere within the environment. In this case, it is highly likely that in order for phishers to cover their malicious activities, they might use DoS as well to cover their tracks in order to avoid detection. This

proves the possibility of DoS and Phishing having relationship. Table 4.8 shows Chi-square test and symmetric measure test result for Operating Algorithm.

Table 4.8: Chi-square test and symmetric measure test result for Operating Algorithm

Findings No.	Relationship between		Chi-Square Test		Symmetric Measure	Outcome
			Pearson Chi-Square value	Probability (p) or significance value	Phi value	
55	Stealth	Polymorphic	11.308	0.001	0.105	There is a relationship
56	Stealth	Anti anti-virus	0.376	0.540	0.019	There is no relationship
57	Polymorphic	Anti anti-virus	18.4	0.000	0.134	There is a relationship

From Table 4.8 above, which shows the statistical results of operating algorithm's sub-feature, it was found that there is a relationship between stealth and anti anti-virus with polymorphic. On the contrary, stealth has no relationship with anti anti-virus because anti anti-virus does not steal anything unless it works against anti-virus programs.

The experimental results of chi-square test and symmetric measure for infection, activation, payload and operating algorithm are presented in Table 4.5, 4.6, 4.7 and 4.8 respectively. Based on the three statistical analyses presented in this study, it is concluded that every relationship has its own interpretation and representation. The relationship between each sub-feature was assessed using Chi-Square tests and symmetric measure in order to prove the relation to each other. From all experimental results presented in Table 4.5 to 4.8, it could be concluded that relationship depends on the nature and the outcome or action of sub features.

4.6 Summary

In this chapter, a new Worm Classification and Detection Technique for Cloud Computing known as EGA is formed and presented based on the research and testing that have been conducted in controlled laboratory environment. The classification consists of five main categories which are: Infection, Activation, Payload, Operating Algorithm, and Propagation. Statistical analysis results show the relationship of various cloud worm features. For infection, activation, payload, operating algorithm and propagation features, this chapter presented the analysis and experimental results for various sub-features. Based on the statistical analyses presented in this research, it can be concluded that every relationship has its own interpretation and representation. The relationship between each sub-feature was assessed using Chi-Square tests and symmetric measure in order to prove the relation to each other. Next, these results are used for EGA cloud worm detection system. This research strongly believes that the integration of the results with the data mining helps to find hidden cloud worm patterns. This improves worm detection accuracy rate in a cloud computing environment which is to be explained in the next chapter. The next chapter introduces GA implementation in cloud worm classification by employing basic feature of GA algorithm.