Detecting Spam Relays by SMTP Traffic Characteristics Using an Autonomous Detection System

By

Hao Wu

A Doctoral Thesis submitted in partial fulfillment of the requirements for the awards of Doctor of Philosophy of Loughborough University

November 2011

Dedicated to

My parents and my family

Abstract

Spam emails are flooding the Internet. Currently, over 90% of emails are spam in the network. Spam emails cost people, ISPs and online services additional money and time, degrade the networks' performance, cause the consumptions of computing and network resources, and cause security problems in networks.

Research to prevent spam is an ongoing concern. A lot of anti-spam techniques have been developed and employed to identify and block spam emails in the network. The commonly used anti-spam email technologies and equipments are DNS-based blackhole lists, content filters, cost based systems, check-sum filters, ham passwords, heuristic filters, honeypots, and so on. However, all the work is not enough, and anti-spam fighters are losing the ground.

SMTP traffic was collected from different sources in real networks and analyzed to determine the difference regarding SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays. It is found that SMTP traffic from legitimate sites and non-legitimate sites are different and could be distinguished from each other. Some methods, which are based on analyzing SMTP traffic characteristics, were purposed to identify spam relays in the network in this thesis.

An autonomous combination system, in which machine learning technologies were employed, was developed to identify spam relays. This system identifies spam relays in real time before spam emails get to an end user. The information that is used to identify spam relays never involves email real content in this system. A series of tests were conducted to evaluate the performance of this system. The results obtained from tests show that the system can identify spam relays with a high spam relay detection rate and an acceptable ratio of false positive errors.

Acknowledgement

I would like to thank Prof. David. J. Parish, my supervisor, for his continual support and encouragement. He gave me directions, motivation, help and advice every time I needed. And he is always kind and understands the problems and limitation of the students. It is only with his support and suggestions that I was able to complete this research.

A special thank-you is given to Dr. John Whitley, whose technical expertise has alleviated many problems. He gave me lots of ideas and helps about the knowledge of TCP/IP networks, the Email systems and the programming. I would like also to my research colleagues in the High Speed Networks (HSN) group for their support and help.

Many thanks are for Dr. Peter Sandford and Dr. Matthew Cook. These two nice persons give lots of help for the data collection in real networks.

Many thanks go to my parents for their encouragement and support. I hope that I fulfill their ambitions.

Thanks for these people who had given me helps and support in the process of this research.

Abbreviations and Acronyms

UBE	Unsolicited Bulk Email
UCE	Unsolicited Commercial Email
SMTP	Simple Mail Transfer Protocol
ISP	Internet Service Provider
MTA	Mail Transfer Agent
POP	Post Office Protocol
IMAP	Internet Message Access Protocol
IPSec	Internet Protocol Security
TLS	Transport Layer Security
DNSBL	DNS-based Block List (DNS-based Blackhole List)
RHSBL	Right Hand Side Blackhole List

URIBL	Uniform Resource Identifier Blacklist
DNSWL	DNS-based White List
SVM	Support Vector Machine
A&R	Authentication and Reputation
HTML	HyperText Markup Language
DCC	Distributed Checksum Clearinghouse
RPD	Recurrent Pattern Detection
CAPTCHA	Completely Automated Public Turing Test to Tell Computers and Humans Apart
MAPs	Mail Agent Providers
K-S Test	Kolmogorove-Smirnov Test

Table of Contents

Abstract ……..... I

Acknowledgement……. II
Abbreviations and Acronyms ……..... III
Table of Contents…….. V

Chapter 1: Introduction……...…… 1
1.1 What are Spam Emails? 1
1.2 Harm of Spam Emails ……..…… 3
1.3 Spam Activity …… 5
1.4 Battles with Spam Emails ……..…….. 7
1.5 Challenge of Anti-spam emails Issue ……..……. 9
1.6 Contributions to Anti-spam Issue ……..……... 11
1.7 Thesis Organization ……..…… 13
1.8 Summary ……...…… 15

Chapter 2: Background and Related Work …….. 16
2.1 SMTP Transfer Protocol and Extension ……...…… 16
2.2 Anti-spam Technology ……. 18
2.2.1 Techniques of End Users …… 18
2.2.2 Automatic Techniques for Email Administrators ……... 18
2.2.3 Techniques of Senders …… 28
2.2.4 Summary of Anti-spam Techniques ……... 29
2.3 Typical Pattern Recognition System …… 30
2.3.1 Sensing ……... 31
2.3.2 Segmentation …….. 32
2.3.3 Feature Extraction ……... 32
2.3.4 Classification …….. 33
2.3.5 Post-processing ……... 33
2.4 Summary …….. 34

Chapter 3: Data Collection ……..…… 35
3.1 TCP/IP Header Structure and 3-way Handshake and Tear Down Protocol …… 35
3.1.1 TCP/IP Header Structure ……..….. 36
3.1.2 3-way Handshake and Tear Down Protocol ……..…… 36
3.2 SMTP Traffic Data Collection ……. 38
3.2.1 Data from a National ISP's Network…….. 38
3.2.2 Data from University Email Servers ……..…….. 39
3.3 Summary 42

Chapter 4: SMTP Traffic Characteristics of Legitimate Email Clients, Legitimate Email Servers and Spam Relays ……...…… 43
4.1 Related Work 44
4.2 SMTP Traffic Characteristics of Legitimate Email Clients ……..… 47

4.3 SMTP Traffic Characteristics of Legitimate Email Servers …… 50
4.3.1 Volume of Connections …….. 50
4.3.2 Ratio of FIN/SYN Flag Set ……. 52
4.3.3 Payloads of Emails on Servers ……... 54
4.3.4 Patterns Related to Time ……. 55
4.3.5 Ratio of Out/In SMTP Packets with SYN Flag Set ……. 58
4.4 SMTP Traffic Characteristics of Email Spam Relays ……. 60
4.5 Difference in SMTP Traffic between Legitimate Users and Spam Relays……..... 66
4.5.1 Evaluate the Successful Connection Rate via FIN/SYN Ratio ….. 67
4.5.2 Count the Total Number of Connections in a Particular Time Interval ……... 67
4.5.3 Compare the Size of the Payload in each Connection …… 68
4.5.4 Evaluate the Ratio of Out/In SMTP Packets with SYN Flag Set. 68
4.5.5 Evaluate the Relationship between the SMTP Traffic, Time of Day and Human Habits (Human Actions) ……... 68
4.6 Summary 70

Chapter 5: An Autonomous System for Detecting Spam Relays by Using SMTP Traffic Characteristics ……..…….. 71
5.1 Components in the Autonomous System ……. 71
5.2 Five Parts of the Autonomous System ……. 73
5.2.1 Sniffer ……. 73
5.2.2 Pre-Processor …….. 74
5.2.3 Trigger …… 80
5.2.4 Classifier ……. 81
5.2.4.1 Algorithms in Classifier ……. 82
5.2.4.2 Final Decision Scheme …….. 88
5.2.5 Post-Processor …… 90
5.2.5.1 Generating Spam Relay Database …… 90
5.2.5.2 Generating Parameters and Thresholds ……. 93
5.2.5.3 Automatic Data Update in System...…….. 97
5.2.5.4 Manual Data Update in System ……. 99
5.3 Autonomous Detection System Structure …… 100
5.4 Summary …….. 103

Chapter 6: Testing and Results …….. 104
6.1 Training Process of the Autonomous System …….. 104
6.1.1 Training Data Set in Training Process ……..….. 104
6.1.2 Training Process ……..…..... 105
6.2 Test Data Sets and System for Tests…… ……. 108
6.2.1 Test Data Sets ……. 108
6.2.2 Autonomous System for Tests …… 109
6.3 Testing Processes and Results ……. 111

6.3.1 Evaluating the Ability to Detecting Spam Relays……... 111
6.3.2 Accessing the Performance of Each Algorithm in the Classifier ... 113
6.3.3 Effect of the Percentile Value for Thresholds on the Performance of the System ……... 117
6.3.4 Performance of the Update Process in the Proposed System 118
6.4 Conclusions of System Tests …….. 120

Chapter 7: Conclusions and Further Work ……..…… 122
7.1 Results and Conclusions …….. 122
7.2 Summary of Contributions …….. 126
7.3 Further work …… 128

Reference …… 129

Appendix 1: Thresholds for Trigger System after Training Process (Percentile Value = 95%)……..……. 145
Appendix 2: Thresholds and Weight values for the Detection System after Training Process (Percentile Value = 95%) ……..... 146
Appendix 3: Results of Test Processes……... 150
Appendix 4: Results from System with Several Algorithms Enabled ……... 158
Appendix 5: Thresholds and Weight values after Update Process …… 161
Appendix 6: Results of Test Processes on System after Update Process 166

Chapter 1: Introduction

This thesis is about detecting email spam relays by SMTP traffic characteristics in Networks. Definition of spam emails and related work are reviewed in this thesis. SMTP Traffic, which collected from real networks, is analyzed to determine the differences regarding the SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays. Also, an autonomous system for detecting spam relays is designed and tested.

In the following paragraphs, the definition of spam emails is presented followed by harm of spam emails and spammers' activity. The last sections discuss the battles with spam emails, challenge of anti-spam emails issue and the contributions of this research.

1.1 What are Spam Emails?

The 21st century is an information century and a network century. The Internet is playing more and more important roles in human lives. But spam emails are harassing people every day through the Internet. There were about 262 billion spam emails sent by spammers per day in 2010 [1]. Only 3% of email is the stuff we want [2]. There are a lot of costs associated with the email spam, including the cost of lost productivity, user education, security problems, network-infrastructure loads, and the development of anti-spam technologies [3].

What is a spam email? A spam email is commonly defined as an unsolicited bulk email (UBE), an unsolicited commercial email (UCE), or a junk email [4] [5] [6]. They are

the emails that not asked for (unsolicited) and received by multiple recipients (bulk). So a spam email should meet the following three characteristics [7]:

1. A spam email is unsolicited.

2. A spam email is a part of a “mass mailing”.

3. The sender of a spam email is a stranger to the receiver.

1.2 Harm of Spam Emails

Email is currently one of the cheapest and most convenient ways to delivery information to others [8]. It is a very efficient sending mechanism, because it could send your information all over the world in one second. Spam emails are flooding networks，simply as a spammer can make a profit or achieve its selfish purposes by sending spam emails. However spam emails are doing harm to others, both networks and human society. Spam emails cost people time and money, cause the consumptions of computing and network resources, degrade the network performance, and lead to a lot of security problems from the networks.

1. Spam emails cost people, ISPs and online services additional time and attention to dismiss these unwanted message. Every day spam emails can be found in mailboxes. They could be business advertisements or activity information. Even worse they could be computer or network viruses. They occupy the room in the mailbox, and it takes time to deal with them.

2. Spam emails cost people, ISPs and online services a lot of money. A recent study by Nucleus Research Inc. reports that the management of spam costs U.S. business owners well over $71 billion per year in lost productivity - that translates to $712 for each employee [9]. Every year, billions of dollars are also spent on the additional equipments, software, and manpower needed to combat the problem.

3. Spam emails cause the consumption of computing and network resources. They degraded the networks' performance. They not only consume the widthband of the networks and reduce the networks' effective transmit speed, but also do harm to the machines in networks. More than 97% of all emails sent over the network are unwanted, and the global ratio of infected machines was 8.6 for every 1,000 uninfected machines [10].

4. Spam emails lead to serious security problems. It is one of the most popular ways to deliver computer viruses and other attaches by using spam emails. Most viruses were broadcasted by sending spam email over the network. Also a lot of crime is related to spam emails. You could be involved in a crime as a victim, because you respond a spam email.

People have to always keep an eye on spam emails. They are bad for our lives and society. Nobody could forecast how many and how serious would be the problems they would cause in the future. But it is certain that we can’t stop fighting with the spam.

1.3 Spammer Activity

Spammers are these hosts, which send spam emails to other hosts in the network. The following sections will introduce how a spammer works.

Firstly, spammers have their own ways to harvest email addresses from anywhere in the network such as web pages, mail lists, chat rooms, UseNet and so on. Spammers will send spam emails to these destination mail addresses including those have been harvested and those which look like they exist or are used [11].

Secondly, Spammers always try to compose messages that are more likely to capture the recipients' attention in order to entice recipients into opening the spam emails. And spammers also try to avoid certain keywords and the phrases that are included in the majority of spam because of the increased use of automated anti-spam filtering tools [12]. But most spam emails in an outbreak from a spammer have similar information. If too many changes are made in each spam email, it will increase the cost of sending the spam and reduce the profit of spammers.

Thirdly, Spammers always send spam emails to a lot of different destinations in the network, after they get a large number of email addresses and compose spam emails. Various spam emails tools [13] are used by spammers to make their messages get through. To avoid detection, spammers usually hide the point of origin. Spammers can send spam emails and remain anonymous by using open mail relays, open proxies, botnets, and so on [14].

Fourthly, while researchers try to develop anti-spam technology, spammers are trying to find new ways to send spam emails. Spammers resort to re-routing their e-mails through third party e-mail servers to avoid detection, and exploit the additional resources of these relay servers. In addition to open relays, spam relays are also established on compromised hosts, which enable spammers to change the IP address of

the spam email source. Nearly 80% of all spam is received from mail relays [15]. Once a bot or zombie is installed on a victim computer system, the controller (Spammer) can send commands to deliver spam emails by this bot. Illegal spam emails sent by zombies has increased dramatically in recent years [16]. After many open relays were closed or placed on the blacklists of the other servers, most spam relays are established on compromised hosts in networks, which enable spammers to change the IP address of the spam email source.

Spammers never think about receivers and networks. What they want to do is send these emails, to increase their profits or for their selfish purposes.

1.4 Battles with Spam Emails

Both end users and administrators of email systems have used various anti-spam techniques to prevent email spam. Most anti-spam techniques can be broken into three broad categories according to the operators: anti-spam techniques used by email end users, anti-spam techniques used by e-mail administrators, and anti-spam techniques used by e-mail senders. There are also some anti-spam techniques that are only employed by researchers and law enforcement officials [17]. Some of these techniques have been employed into products, services and software to identify and block spam emails in the network.

The commonly used anti-spam email technologies and equipment are DNS-based Black Lists, Content Filters, Statistical Filters, Cost Based Systems, Check-Sum Filters, Authentication and Reputation (A&R), Sender-support Whitelists and Tags, Ham Password, Heuristic Filters, Honeypots, Hybrid Filtering, Outbound Spam Protection, PTR/Reverse DNS Checks, SMTP Callback Verification, Egress Spam Filtering, Spam Report Feedback Loops, and so on. [18][19][20][21][22][23][24]

A large number of techniques have been playing an important role in the war of anti-spam emails. But the general consensus is that a single technical solution that is able to prevent the propagation of spam is unlikely to be found given the constraints of the current Internet architecture [25]. And each has trade-offs between spam detection rate vs. false positive error rate, and the trade-offs between the associated costs and effort. So a composite approach that applies many of techniques introduced above could be more helpful to solve the problem and reduce the amount of the spam emails.

In 2004 Bill Gates claimed, “Spam will be a thing of past” [26]. But in Microsoft's biannual report on the state of computer security in 2009, the company said that over 97.3 percent of email traffic was unwanted spam in the second half of 2008 [27].

Figure1.1 shows the percentage of spam in email in August 2010. [28]

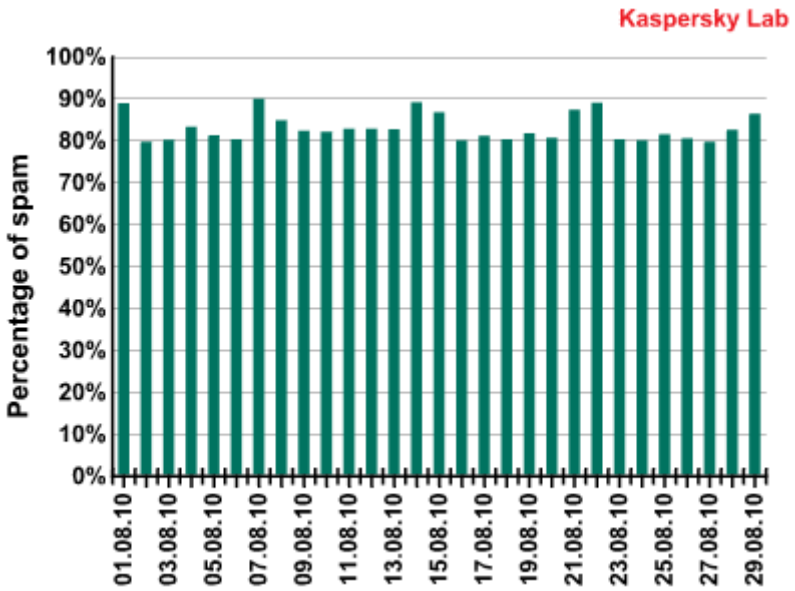

Figure 1.1： Percentage of Spam in Email in August 2010

The amount of spam detected in mail traffic averaged 82.6% in August 2010. A low of 79.4% was recorded on 2nd August 2010, with a peak value of 89.7% being reached on 7th August 2010. In another statistical report from Computer Services of the Loughborough University, it has been shown that the percentage of the emails that have been flagged as junk and rejected has increased by a dramatic amount [29].

A lot of time and resources have been spent on developing and applying anti-spam technology. But the anti-spam fighter is losing ground before the spam with dramatic speed.

1.5 Challenge of Anti-spam Email Issue

Spam emails have been flooding over the network since email became one of popular communication methods. People tried to fight back by producing all kinds of anti-spam emails techniques.

Currently, more and more money and manpower have gone into the anti-spam actions. But anti-spam fighters still keep on losing ground before spammers. Why is the situation becoming more serious? There are maybe three main reasons, which involve not only the limitation of anti-spam techniques but also human social issues.

1. Spam fighting is an unbalanced war. It is easy for spammers to find a way to break these anti-spam systems. But it is difficult for spam fighters to find a way, which is a good way which does not harm other legitimate users, to fight back.

2. Spam is a fascinating topic. Dealing with spam mails involves several fundamental rights, including free speech, privacy, private property and freedom of association [30]. It also raises some remarkably strong emotions. It is difficult to deal with these cases by strict laws. Little support from laws and human society make the problem difficult and serious.

3. Targets of the anti-spam actions lost their way. Targets of most anti-spam techniques are protecting email end users. The majority of anti-spam systems focus on filtering the spam email at the end-users' terminals. It is helpful to prevent the legitimate email users from receiving spam emails, but fighting with spam can't be only on the end-users. More anti-spam techniques, which detect and block spam emails at different stages, need to be developed and take part in anti-spam fighting.

In the future, more powerful combination anti-spam methods should be researched and produced. Anti-spam email fighting should occur not only in the receiving email stage but also in the transmitting email stage. The fight should not only be in technical

areas, but also in human social areas including law and morality.

Legitimate users, networks and human society require the stopping of the spam emails. Today spam emails are doing harm to every area of human lives. Spam email is a big problem because of the shared and private resources it consumes; Spam email is a big problem because of the large number of victims it involves; and spam email is a big problem because of the difficulty of getting rid of spam in the network. So we can never stop the fight with spam emails.

1.6 Contributions to the Anti-spam Issue

1. The first contribution of this thesis is the collection of SMTP traffic data from real networks. There are some SMTP traffic datasets in the public domain for download. But we didn't use any dataset in the public domain in this thesis, because we do not know any detail about the network in which these datasets collected. SMTP traffic data in this thesis are from two networks: One is a nationwide ISP's local network, and the other is the Loughborough University campus network. A sniffer was created in the C programming language to collect SMTP traffic data from the Loughborough University campus network. We have also access to a national ISP's traffic which had been previously collected by Dr Peter Sanford. SMTP traffic was from not only the good sources (legitimate email clients and legitimate email servers) but also the bad sources (spam relays). SMTP traffic data was used for analysis to determine the differences regarding SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays.

2. The second contribution of this research is the analysis of SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays. Legitimate sites (legitimate email clients and servers) and illegitimate sites (spam relays) have been shown to have their own SMTP traffic characteristics in this research, and can be distinguished from each other by using SMTP traffic characteristics. The understanding of the SMTP traffic characteristics of different hosts (legitimate email clients, legitimate email servers and spam relays) suggested some methods, which might be possible to be used to identify spam relays in networks. The methods are evaluating the successful connection rate by the FIN/SYN flag set, counting the total number of the connections in a particular time interval, comparing the size of payload in each connection, evaluating the ratio of Out/In SMTP packets with SYN flag set, and evaluating the relativity between SMTP traffic and time of day.

3. The third contribution of thesis is in developing an autonomous system for detecting

spam relays by SMTP traffic characteristics. Six algorithms, which are correlated to the methods suggested by the analysis of SMTP traffic characteristics, are combined in the classifier of the system. The results from the tests show that this proposed system has a good performance of spam relay identification. Over 90% of spam relays could be identified by this system, and the rate of false positive errors is about 0.13 % on average.

4. Spam relay identification in this system avoids infringing upon the rights of people's privacy, because it never involves reading the email real content. Only the TCP/IP header information of the SMTP packets are logged and used for detection in this system.

5. This system is able to identify spam relays on the spam's transmit stage. It is better to improve the performance of the network than remove the spam emails at the receivers' terminals.

6. The system can be adjusted by network administrators to achieve a satisfactory performance, which meets to the requirements of the network management. Changing of percentile value for thresholds in this system has been shown to affect the performance. Setting this percentile value can help administrators to achieve a satisfactory performance.

7. In this thesis, it is also shown that a combination system could have a better performance of spam relay identification. Although each individual algorithm combined in the classifier is able to pick out a number of spam relays; a system, in which more algorithms are combined, can have more opportunities to provide better performance.

1.7 Thesis Organization

The thesis is organized in the following way:

Chapter 1 (this chapter) gives an insight into the research work. It reviews the definition of spam emails, and discusses the harm of spam emails and spammer activities. It also introduces the battles with the spam emails and the challenge of anti-spam issues. Finally, the contributions of this research are presented.

Chapter 2 explores the background and related work. It reviews the protocol of SMTP, and explains how the disadvantages of SMTP lead to the spam email explosion. A larger number of anti-spam techniques (e.g. DNSLs, Content Filters, Bayesian Spam Filter, Checksum Based Filters, and so on) commonly used in networks are introduced. Finally, the structure of a typical pattern recognition system is presented.

Chapter 3 introduces the collection of SMTP traffic data. Firstly, the TCP/IP header structure and 3-way handshake and teardown protocol are reviewed to help understand the process of collection. Secondly the process of SMTP data collections from a commercial ISP's network and Loughborough University campus network are introduced in detail.

Chapter 4 is dedicated to the analysis of SMTP traffic data, which has been collected. In Chapter 4, SMTP traffic is analyzed to determine the differences regarding the SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays. It states that the SMTP traffic characteristics of legitimate sites and illegitimate sites are different and can be distinguished from each other. In that chapter, it is also suggested that some automation methods may be used to identify spam relays in the network.

Chapter 5 of this thesis proposes an autonomous system, which identifies spam relays by using SMTP traffic characteristics. In Chapter 5, the proposed system is described in

detail including the components and identification mechanism. Finally, the detection process of the system is explained step by step by using the flow chart of the system structure.

Chapter 6 discusses the training and tests of the system. The training process of the system is introduced in this chapter. A series of tests have been conducted to evaluate the performance of the detection system. Results obtained from the series of tests are presented in this chapter. In Chapter 6, it is also shown that each individual algorithm gives a contribution to the spam relay identification; however, a combination system can provide a better performance. In this chapter, it indicates how the percentile value for thresholds affects the performance of the system by using test results. Finally in Chapter 6, test results indicate that the update process in the system works well in keeping the performance of spam relay identification as good as expected.

Chapter 7 summaries the conclusions and gives glances of future research work.

1.8 Summary

Spam emails are unsolicited bulk emails. They do harm to people's lives and society. A lot of money and manpower have been invested in anti-spam actions, but people still keep losing ground before spam. We can never stop the fight with spam email.

The objective of this research work is to determine the differences regarding the SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays, and to develop an autonomous system for detecting spam relays by using SMTP traffic characteristics.

Chapter 2: Background and Related Work

In this chapter, background and related work about this research are introduced. Firstly Simple Mail Transfer Protocol is reviewed, and it also indicates that the disadvantages of SMTP lead to the spam emails explosion. Secondly, a lot of various anti-spam techniques commonly used in networks are introduced in details. At last in this chapter, the structure of a typical pattern recognition system is presented, which is able to help to design an autonomous system involving machine learning technology for detecting spam relays in the network.

2.1 Simple Mail Transfer Protocol and Extension

Simple Mail Transfer Protocol was originally used to exchange text messages between nodes on the United States Department of Defense's Defense Advanced Research Projects Agency (DARPA) internetwork. Currently, it is widely used as an Internet standard for electronic mail transmission across Internet Protocol (IP) networks. SMTP was first defined in RFC 821 (STD 15) (1982) [31], and last updated by RFC 5321 (2008) [32], which included the extended SMTP (ESMTP) additions.

SMTP can be used to send and receive emails by email servers and other mail transfer agents (MTA). However, most time user-level client applications only use SMTP for sending emails to a mail server for relaying. Clients application usually use either the Post Office Protocol (POP) [33] or the Internet Message Access Protocol (IMAP) [34] to access their mail accounts on a mail server for receiving email messages. Sometimes a system (e.g. Microsoft Exchange) is used to receive email messages from an email server by email users. SMTP is specified for outgoing mail transport and uses TCP port 25.

SMTP has aided in the widely using of emails in the network, but has contribution to the abuse as well. The abuses of email cause that spam emails are flooding Internet. As one of the most important transmission protocols in the network, SMTP was intended to be simple and robust. It was designed to be open and use human-readable commands. Relaying is necessary for a robust message delivery. Therefore, relaying is a legitimate function of SMTP. Relaying enables an MTA to send an email message to the nearest available MTA if the intended receiving MTA is offline and unreachable. But abuses of relaying generate vast spam emails, which are doing harm to both people and networks. SMTP proxies (also known as SMTP application-level gateways) are popularly used to transmit emails across network boundaries. Similar to relays, proxies can also be used for sending spam if they are not properly secured. [35]

It would be very difficult to replace SMTP outright because of the global acceptance and reliance. Therefore, it is important to find ways to prevent abuse of relaying service, which causes mass spam emails in the network. Address restrictions, SMTP authentication and some network-security mechanisms (e.g. IPSec, TLS) can be used to limit accesses for preventing the abuse of relaying. Also many anti-spam techniques have been developed to identify and block spam emails in the network. The following sections will introduce some commonly used anti-spam techniques.

2.2 Anti-spam Technology

Currently, using an anti-spam technical solution is the most effective and commonly used means to prevent spam emails. A variety of methods have already existed, but each with its respective merits and disadvantages. Most anti-spam techniques can be divided into three categories according to the operators: End-User Techniques, Automatic Techniques for Email Administrators, and Automatic Techniques for Email Sender (such as MTA, email servers).

The following subsections are to introduce several commonly used anti-spam techniques.

2.2.1 Techniques of End Users

There are a number of techniques which email end users can use to reduce and prevent spam emails. The most popular methods are about restricting the availability of their email addresses to spam. Disposable email address, discretion sharing the email addresses only in limited groups, and avoiding responding to spam could help to avoid the spam email addresses harvesting. These techniques could help to prevent spam email. Also, people could report spam emails to the anti-spam service on networks, such as a network abuse clear-house. This can help the network administrators to terminate the spam services.

2.2.2 Automatic Techniques for Email Administrators

A number of appliances, services, and software, which can be used by email administrators, have been employed to reduce spam emails on email systems and mailboxes. Some of these (e.g. DNSBLs) reduce spam by rejecting emails that are from those sites known or likely to send spam emails. Other more advanced techniques are able to detect spam emails by analyzing message patterns in real time. Machine

learning techniques, which can improve accuracy over manual methods, are popularly employed in many anti-spam filtering systems. The following subsection is to introduce several commonly used automatic anti-spam techniques for email administrators.

DNSLs

DNSLs are DNS-Based Lists. DNS-Based Lists Anti-spam Systems list good (white) or bad (black) IPs or URLs, including RHSBLs and URIBLs [36].

A system listing the good sources could be named DNSWLs (DNS-Based White Lists System). In this kind of systems, the emails coming from these sources that have been listed will be passed at any time in any situation. The most obvious disadvantage of such a method is that it restricts communication to already established contact, which is impractical for majority of end users. [37]

The most popular used DNSLs anti-spam system is DNSBLs (DNS-based Blackhole Lists). DNSBLs are used to block a series of particular lists (typically of IP address) via the DNS [38]. DNSBLs are popularly used by ISPs and anti-spam service companies to keep track of a group of IP addresses that generate spam emails. The emails sent from these IP addresses that have listed in the system will be rejected out-of-hand. In such a way, the mail servers can easily be set to reject mail from some unwanted sources. These sources could be known as email spammers, spam supporters or spam relay hosts.

These DNSLs systems are the good spam-fighters. A lot of these sources for the lists are available on the Internet [39] [40]. They make decisions by strict rules. But the spammers will normally change the source IPs or use the other hosts to do their jobs. In order to make decisions accurately, these lists should be upgraded as frequently as possible.

Content Filtering

Spam content filters identify spam mails by the nature of the content of each email. Content filtering is commonly implemented by many email end users. It is popularly used to reduce unsolicited bulk Email (UBE), which is most like to contain some predictive keywords. These predictive keywords are used to identify spam emails in content filters [41]. The information, which could be used to detect spam emails, is contained in the mail bodies or on the mail headers (like "subject"). The techniques applied in content filters, are Bayesian Classifier [42][43][44], memory-based approach [45][46], support vector machine (SVM) [47][48][49][50][51], the technique of maximum entropy [52][53], neural networks [54][55][56][57][58], genetic programming [59][60], and so on.

One of the most popular content filters is the Bayesian filter. The Bayesian Spam Filter [42][43][44][45][61] is a statistical technique of email filter, and it makes use of a Naive Bayes Classifier to identify spam emails. Basically, Bayesian-based filtering approach uses the knowledge of prior events to predict the future events. Email messages are marked as spam and non-spam, and Bayesian-based filters can learn to automatically put messages from the same source or with the same kind of patterns into the corresponding category. Bayesian-based filters identify email spam based on some pre-defined tokens (words, phrase or sometimes other things) [62]. Keywords-based Bayesian spam filtering [63] are employed to identify spam emails popularly in the network. Particular words have particular probabilities of occurring in spam emails and in legitimate emails. The probability that an email with a particular set of words is computed by using these word probabilities then is used to identify which category (spam or non-spam) this email belongs to. Only pre-defines keywords give their contributions to the email's probability in most Bayesian-based content filters. This contribution is called the posterior probability and computed using Bayes' theorem [64]. Then, if an email's probability exceeds a certain threshold, the filter

will mark the email as a spam email. The formula used by the software to determine that is derived from Bayes' theorem [65].

$$\Pr(S|W) = \frac{\Pr(W|S) \cdot \Pr(S)}{\Pr(W|S) \cdot \Pr(S) + \Pr(W|H) \cdot \Pr(H)}$$

Where:

- $\Pr(S|W)$ is the probability that a message is a spam, knowing that the typical word is in it;
- $\Pr(S)$ is the overall probability that any given message is spam;
- $\Pr(W|S)$ is the probability that the typical word appears in spam messages;
- $\Pr(H)$ is the overall probability that any given message is not spam (ham);
- $\Pr(W|H)$ is the probability that the typical word appears in ham messages.

The first known Bayes classifier program used to sort mails into folders was Jason Rennie's iFile program released in 1996 [66]. The first scholarly publication on Bayesian spam filtering was by Sahami in 1998 [61]. In 2002 Paul Graham was able to greatly improve the false positive rate, so that it could be used on its own as a single spam filter [67] [68]. Bayesian spam filters are one of the most effective anti-spam techniques, and Bayesian mathematics can be applied to the spam problem. Bayesian spam filters results in an adaptive "statistical intelligence" technique that can achieve a very high spam detection rate and gives low false positive spam detection rates that are normally acceptable to most.[69][70]

Some other common content filters are attachment filters, mail header filters, Language filters, Regular Expression filters, content-encoding filters, HTML anomalies filters, and so on [71].

Usually the content filters are also used for anti-virus protection. They can scan the

binary attachments of the mails or the HTML contents. It is a good way to help to stop the leaking of secret information.

Checksum-Based Filtering

Checksum-based filters detect spam emails based on the fact that the spam messages will be identical with only small variations. Checksum-based filters strip out everything (such as name of receiver, date) that might vary between messages, reduce what remains to a checksum, and look that checksum up in a database which collects the checksums of messages that email recipients consider to be spam. If the checksum is already in the database, this message is likely to be spam.

An advantage of checksum-based filtering is that it lets ordinary email users take part in identifying spam. But spammers can insert unique invisible gibberish (known as hashbusters [72]) into each of their messages, which makes each message unique and has a different checksum. This leads an arm race between the anti-spam developers of the checksum-based software and the developers of spam generating software.

Checksum based filtering methods include Distributed Checksum Clearinghouse and Vipul's Razor, which will be introduced in the following subsections.

- Distributed Checksum Clearinghouse

  Distributed Checksum Clearinghouse (also referred to as DCC) is a hash sharing method of spam email detection [73]. The basic logic in DCC is that most spam emails are sent to many recipients. The email, that has the same message body appearing many times, is therefore a spam email. DDC, as an antis-spam system, is made up of a distributed collection of Clearinghouse (servers), where counts of email messages received by email clients are maintained. DDC identifies spam emails by taking a checksum and sending that checksum to a Clearinghouse. The

Clearinghouse responds with the number of that checksums received. A spam emails can be identified due to its response numbers is high.

DCC is resistant to hashbusters because “the main DCC checksums are fuzzy and ignore aspects of messages. The fuzzy checksums are changed as spam evolves” [74].DCC is likely to identify mailing lists as bulk email unless they are white listed. The content is not examined. DCC works over the UDP protocol and causes some additional network traffic.

- Vipul’s Razor

Vipul's Razor, as a checksum-based, distributed, collaborative system for identifying and filtering spam emails in a distributed fashion, consists of a set of Razor servers that hold the database of known spam. A user’s email client is configured to submit income emails to a razor client that queried a server if it is a known spam. Detection is done with statistical and randomized signatures that efficiently spot mutating spam content. Users can not only add spam emails to the database maintained on the servers, but also can flag emails misidentified as spam in the system. The weight assigned to a given user’s classification is determined by a trust level, which is generated by considering “consensus” of this user’s previous classifications and system’s. [75]

Vipul's Razor was written in Perl by (primarily) Vopul Ved Prakash. Razor is not only used directly by some email clients, but also used by some server-side spam filters, for example SpamAssassin [76]. And a commercial derivative of Razor, named Cloudmark Authority, is available from Cloudmark [77].

Statistical Filtering

Statistical filtering was first proposed in 1998 by Mehran Sahami, at the AAAI-98 Workshop on Learning for Text Categorization [61]. And statistical filtering was

popularized by Paul Graham's influential 2002 article A Plan for Spam. Based on collections of spam and non-spam ("ham") email submitted by users [67], that article proposed the use of naive Bayes classifiers to predict whether messages are spam or not.

Statistical content filtering doesn't need to require maintenance per second after it is set up. The system users mark emails as spam or non-spam. And the filtering software collects these judgments and keeps these judgments as records for detecting the spam emails. A statistical content filter is a kind of document classification system, and a number of machine learning research have turned their attention to this direction. Machine learning technologies employed in statistical filters not only make anti-spam statistical filters responding quickly to the changes of spam without administrative intervention, but also improve the performance of identifying spam.

Not only natural real contents of emails are looked at in statistical filtering, also email message headers can be considered. Thereby, statistical filters identify spam emails also by considering peculiarities of the transport mechanism of the email. Some algorithms in statistical filtering involve email inter-arrival times, email size, number of recipients per email and so on. Characteristics of spam traffic and spammers are also widely used in statistical filtering to identify spam email.

Software programs implementing statistical filtering include Bogofilter, the e-mail programs Mozilla and Mozilla Thunderbird, and later revisions of SpamAssassin [78]. Another interesting project is CRM114 which hashes phrases and does Bayesian classification on the phrases [79].

Cost Based Systems

Low cost is one of the important reasons for the spammers to broadcast the information by spam mails. Each spam email cost spammer less than $ 0.00001 on average [80]. The

cost factors for a spammer can be grouped in four categories—hardware cost H, software cost S, operating cost O, and labor cost L. So a basic cost model for spammer can be defined as: Total cost C= H + S + O + L [81]. So most spam-fighters advised to use cost based systems to increase the spammers' cost.

One of the cost based plans is the stamp system. The sender will pay electronic money to the recipient, or the ISP, or some other gatekeeper. The spammers will spend a lot of electronic money on the spam sending. Also, it provides another way to point out the spammers by counting the stamps used by the sender in their account. A refinement to stamp systems is that the method of requiring that a micropayment only be made if the recipient considers the email to be abusive. Therefore in stamp systems, popular free legitimate mailing list hosts would be unable to continue to provide their services if they had to pay postage for every message they sent.

Another plan is the Proof-of-work systems and similar systems [82][83]. They ask the sender to pay a computational calculation cost, which will take the sender several seconds per email. But for a spammer, the millions of spam mails that he sent will cost him a long time. The large number of calculations will slow down the spammer's computer [84]. The point is to slow down hosts that send mostly spam-often millions and millions of them. While a user that want to send email to a moderate number of recipients suffers just a few seconds' delays, sending millions of emails would take an unaffordable amount of time. Proof-of-work such as Hashcash and Penny Black require that a sender pay a computational cost by performing a calculation that the receiver can later verify. Verification must be much faster than performing the calculation, so that the computation slows down a sender but does not significantly impact a receiver. The disadvantage of these techniques is that they will suffers when the sender maintains a computation farm of their own or used zombies.

These two plans increase the spammer's cost of the money and time. But these systems will also inconvenience legitimate email users. And most users will feel uncomfortable

paying for email sending.

Pattern Detection

Pattern detection is an approach to detect spam emails in real time before they get to an end user. Many spam messages have similar content or may contain similar attachments which this detection technique can catch. Pattern detection techniques identify spam patterns by monitoring a large database of messages worldwide.

Recurrent Pattern Detection (RPD) is one of anti-spam software tools based on pattern detection techniques. This method is developed by Commtouch, a developer of Anti-Spam software. Recurrent Pattern Detection is more automated than most because the service provider maintains the comparative spam database instead of the system administrator. This software can be integrated into other appliances and applications. The following subsection will introduce Recurrent Pattern Detection.

Recurrent Pattern Detection [85]:

Recurrent Pattern Detection (RPD) technology, a patent-pending technology based on Commtouch's U.S. patent, identifies and classifies all types of suspecious patterns of email in real-time by extracting and analyzing relevant email patterns. RPD is hosted by the Commtouch® Detection Center, which proactively analyzes vast amounts of Internet traffic in real-time. Both distribution patterns and structure patterns are able to be classified by RPD. The distribution patterns represent the characteristics of senders (how many, location) and the volume of emails sent over a period of time. The structure patterns are about random combinations of text from the header, and body of the message as well as URLs found to be repeated in different messages [86]. The Analysis results are kept in a database of classifications. RPD can not only be used to identify new suspicious patterns, but also be used to modify or enhance classifications of already identified email patterns. RPD is also designed to distinguish between the patterns of solicited bulk emails ('good' messages such as newsletters, mailing lists,

etc.) from unsolicited bulk emails by applying a reverse analysis.

The RPD technology does not require human intervention and is designed to be fully automated. It can identify new threat outbreaks within minutes since they are generated on the internet. Hashed values of email patterns are analyzed in RPD technology. And email real content is never involved, which ensure maximum privacy and business confidentiality.

Ham Passwords

Some anti-spam techniques by using ham passwords ask unrecognised senders to include a password indicating that this email is a “ham” (not spam) message in their email. It must be made sure that the legitimate senders will be able to find the ham password. Typically people use a web page to give the email address and password that is expected to be used by legitimate senders. The email address could be given as a set of instructions, and the ham password could be given as shrouded graphical image. Generally, this information about email address and password can’t be easy to read by machine. [87]

The ham password may be included in the “subjected” line of an email address. Also it could be appended in the “username” part of the email address, such as the plus addressing technique. Ham passwords not only can be used to identify unauthorized emails, but also are often combined with filtering systems to evaluate the risk that a filtering system will accidentally identify a ham message as a spam message.

Honeypots

A honeypot is a trap set to detect unauthorized use of the information systems. Honeypots help to improve the overall security architecture by providing early warning, which is about new attacks, attacking techniques and so on. Honeypots help to monitor attackers as they exploit systems. [88]

Honeypots are being used to reduce spam emails by setting up “trap” email accounts to identify the sources and nature of the emails received [89]. An approach is that setting up an imitation MTA which gives the appearance of being an open mail relay, or an imitation TCP/IP proxy server which gives the appearance of being an open proxy. Spammers who probe systems for open relays/proxies will find such a host and attempt to send mail through it. This “trap” system not only causes spammers wasting their time and resources, but also could collect the sources and nature of spam emails which spammers are sending to the “trap”. Such information collected by the honeyposts could be used to identify the spam. For example, the sources of spam emails that are collected by the honeypots could be submitted to DNSBLs for stop the spam. [90][91]

2.2.3 Techniques of Senders

There are a variety of techniques which emails senders can use to make sure that they do not send spam emails, such as background checks, confirmed opt-in mailing lists, egress spam filtering, rate limiting, limit email backscatter, and so on. The following subsections will represent some of anti-spam techniques of email senders.

Background checks on new users and customers

Most email senders do background checks on new users and customers to avoid their systems being used to send spam emails. CAPTCHAS [92] [93] is popularly used on new account by most ISPs and web email providers to verify that it is a real human registering the account, instead of an automated spamming system.

Confirmed Opt-in Mailing Lists

To prevent spam abuse, all mailing lists are encouraged to use confirmed opt-in (also known as verified opt-in or double opt-in) by MAPs and other anti-spam

organizations. Whenever a new subscriber (an email address) asks to be subscribed to the mailing list, the list software should send a confirmation message to verify it is really them. Confirmed Opt-in mailing list software should send a confirmation message to the address, which is presented for subscription to the list. The confirmation message must not contain any advertising content, so it is not construed to be a spam message itself. New subscriber is not added to the live mail list unless the recipient responds to the confirmation message, such as clicking a special web link or sending back a reply e-mail. [94]

2.2.4 Summary of Anti-spam Techniques

A lot of anti-spam techniques have been used by end-users, email administrators and even legitimate email senders. Some commonly used anti-spam techniques have been represented in the previous sections. These techniques are playing an important role to identify and block spam emails in the network, but all these techniques are not enough. There are still hundreds of billions of spam emails in a year. Spam characteristics and lots of detection methods are employed in anti-spam techniques. However, there is still not a single technique that is able to prevent the propagation of spam in current networks. In this thesis, we try to find different SMTP characteristics among email user clients, legitimate email servers and spam relay hosts, and build an autonomous system to detect the spam relays via their SMTP traffic characteristics.

2.3 Typical Pattern Recognition System

In this research, we try to build an autonomous system, which employs machine learning techniques, for detecting the spam relays by using SMTP traffic characteristics in the network. Machine learning is one of the branches of artificial intelligence. It is a science of developing algorithms that allow the machine to make inferences from observing data (empirical data, such as from sensor or databases), generalize it to rules and make predictions on attributes or future data. Currently, machine learning technologies are widely used for data mining, autonomous discovery, data updating, programming by example, etc. [95]

Classification, which is also referred to as pattern recognition, is one of the important tasks of machine learning techniques. Pattern recognition techniques are employed in a proposed system for detecting spam relays in this research. A typical pattern recognition system usually includes 5 parts: sensing, segmentation, feature extraction, classification and post-processing. **Figure 2.1** is the slightly more elaborate diagram of the components of a typical pattern recognition system. [96]

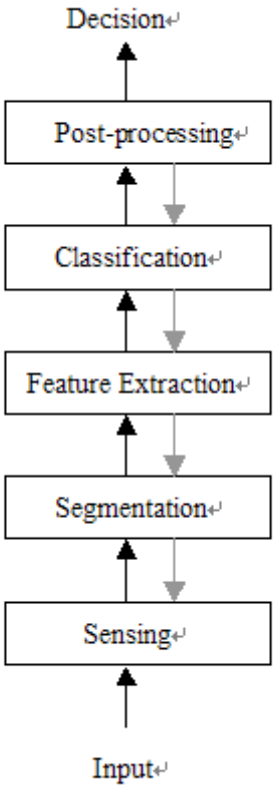

Figure 2.1: Structure of a Typical Pattern Recognition System

2.3.1 Sensing

The Sensing element of a pattern recognition system gets inputs from the environment or object, which is being monitored. The most important work in design of sensor is to choose which information should be collected from the monitored object. A good design of sensors for a pattern recognition system helps to collect the exactly original information, which can improve accuracy of system's output.

The difficulty of the problem may be well depend on the characteristics and limitations of the sensor, such as its bandwidth, resolution, sensitivity, distortion, signal-to-noise ration, latency etc.

2.3. 2 Segmentation

Segmentation is one of the deepest problems in pattern recognition. Individual patterns have to be segmented for pattern recognition system to use as inputs. A way must to be found when we have switched from a pattern to another.

2.3. 3 Feature Extraction

A feature extraction is used to characterize an object to be recognized by measurements in a system. It picks up the distinguishing features, whose values are very similar for objects in the same category and very different for the objects in different categories, and passes them to a recognition system for identification.

Different feature extraction methods are designed for different representations of characteristics, such as solid binary characteristic, character contours, and skeletons or gray level sub-image [97]. Texture and Shape features are widely used for the analysis in the area of image processing. In this area, feature extraction methods commonly used include statistical grey level features, histogram features, the surrounding region dependence method, GLCM features, grey –level difference methods, axis of least inertia, center of gravity, average bending energy and so on [98][99]. Feature extraction, an essential component in data mining and anomaly detection, also summarizes the behavior from a traffic data packet steam. Features monitored in network data traffic could be one or several from the group of source address, destination address, traffic volume, port, payload, distribution characteristics and so on.[100][101]

A good feature extractor would make the job of the classifier easier. To help the classifier to make the correct decisions, it yields inputs describing the distinguishing features. Selection of a feature extraction method is probably the single most important factor in achieving high recognition performance.

2.3. 4 Classification

The classifier component of a pattern recognition system is used to assign an object to its related category via the distinguishing features provided by the feature extractor. The general task of most classifiers is to determine the probability for each possible category.

It is impossible to find a “perfect” classifier. Performance of a classifier is affected by the difficulty of the classification problem, which depends on the variability (due to the complexity or noise) in the feature values for objects in the same category and the relativity of the difference between feature values in different categories [96]. A single technical solution that is able to stop spam is unlike to be found currently. Spam identification algorithms are combined in a classifier, such as Bayesian classifier, SVM classifier, linear classifier and so on, to fight spam emails. A lot of anti-spam techniques have been introduced in section 2.2.

The simplest measure of classifier performance is the classification error rate, which is the percentage of new patterns assigned to the wrong category. Positive false error rate and Negative false error rate are commonly used to evaluate the performance of a pattern recognition system.

2.3. 5 Post Processing

It is very important for a pattern recognition system to improve the performance automatically in its classification process. The post-processor uses the output of the classifier to evaluate the performance and decide on the recommended action that can be used to improve the performance of system. The recommended actions include updating of the system, generations of thresholds and parameters, and so on. All of these actions help the system to be automatically operated and improve the performance of the system.

2.4 Summary

Currently, Simple Mail Transfer Protocol (SMTP) is widely used for sending and receiving mails by email servers and other mail transfer agents. The abuse of relaying, which SMTP allows for robust message delivery, causes the explosion of spam email. A lot of anti-spam techniques have been developed and employed for identifying and removing spam emails in the network, such as DNSLs, Cost Based Systems, Checksum Based Filters, Content Filters, Bayesian Spam Filter, and so on.

Machine learning technologies are popularly used in autonomous systems. An important task of machine learning is pattern recognition. And a pattern recognition system usually includes 5 parts: sensing, segmentation, feature extraction, classification and post-processing. Each part gives the contributions to the performance of the system. An autonomous system, in which machine learning technologies are employed, was proposed to identify spam relays in the network in this research work.

Chapter 3: Data Collection

In this research, SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays were analyzed with the aim of finding a method to identify spam relays in the network. To do this, SMTP traffic was collected from real networks. In this chapter, the collection of SMTP traffic is described in detail.

The TCP/IP header structure and 3-way handshake & teardown protocol are reviewed in this chapter for helping to understand the process of the SMTP traffic data collection. This review is followed by explanation of SMTP traffic data collections used in this research, including introducing the real networks which the data was collected from, indicating the parameters and flag sets which were recorded in the traffic data collection processes, and presenting the processes of SMTP data collections from the different sources.

3.1 TCP/IP Header Structure and 3-Way Handshake & Tear Down Protocol

Understanding the TCP/IP Header Structure and 3-ways Handshake & Tear down Protocol can help explain why and how we collected these parameters and flag sets from the packet headers of SMTP traffic.

3.1.1 TCP/IP Header Structure

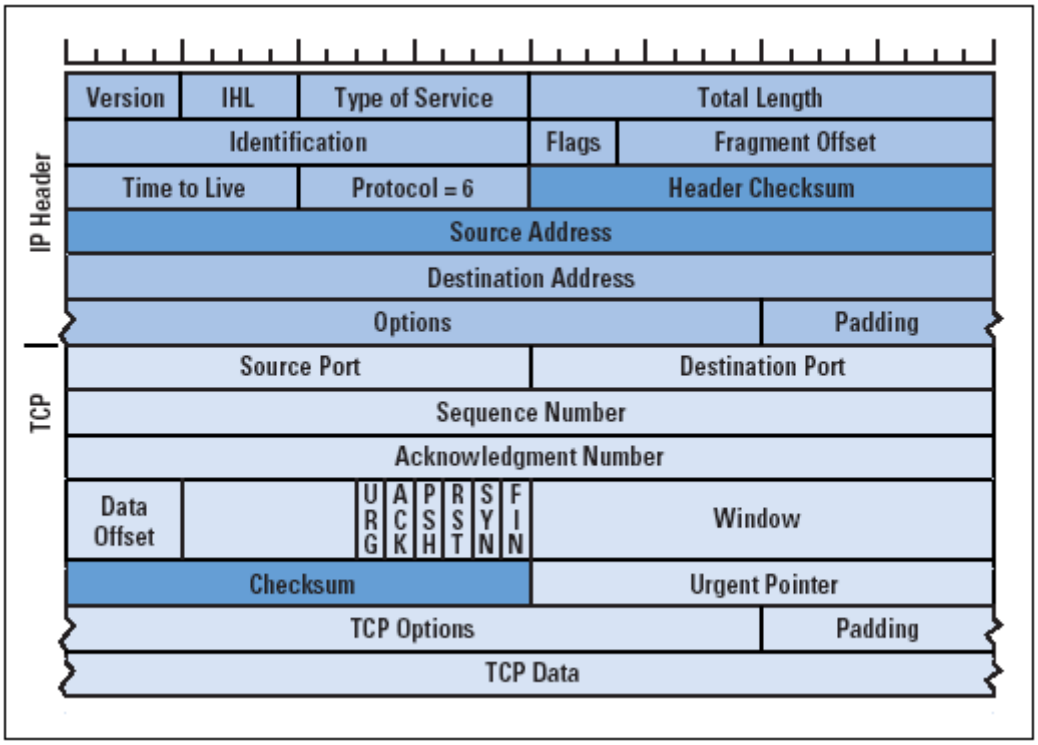

Figure 3.1：TCP/IP Header Structure [102]

Figure 3.1 shows the TCP/IP header structure. From the information of the packet' TCP/IP header Structure , it is easy to got the packet's Total Length, Source Address (Source IP), Destination Address (Destination IP), Sequence Number, Source Port, Destination Port, States of the TCP flags, and so on. All of the SMTP traffic packets have a destination port value of 25.

3.1.2 3-Way Handshake & Tear Down Protocol

The three-way handshake in the Transmission Control Protocol, also called the three-message handshake, is the method used to establish and tear down the network connection. This TCP handshaking mechanism is designed so that two computers attempting to communicate can not only negotiate the parameters of the network connection before beginning communication but also establish separate connections at

the same time. [103]

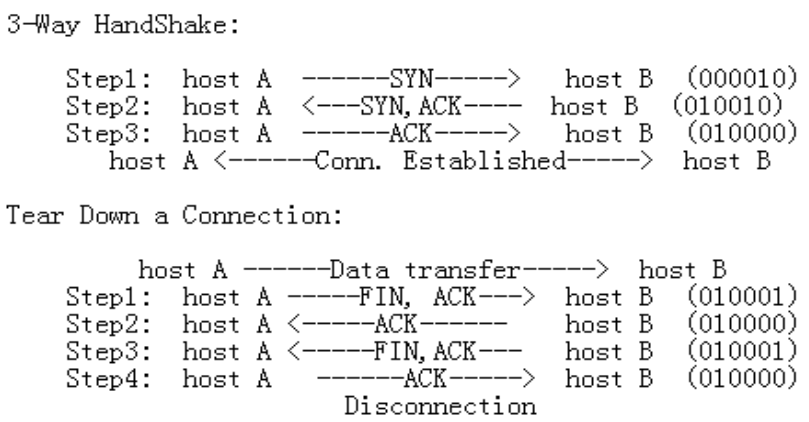

Figure 3.2: TCP Three-way Handshake and Tear Down Protocol

The process of the TCP three-way handshake and tear down is shown in the **Figure 3.2**, and the following characteristics can be determined:

1. The outgoing SMTP packets with **SYN**chronize flag set are closely correlated to the TCP connections which the host tried to establish.

2. Packets with **FIN**ish flag set are correlated to the number of completed connections.

3. The size of the payload for each connection is related to the contents which have been sent.

So in the process of SMTP data collection, Capturing times, Source IP, Destination IP, States of flags (SYN, FIN, ACK, RST), and size of payload of each captured packet are recorded. These parameters related to the characteristics previously described. A summary of the volumes of packets with specific flag set in a particular time interval set is also made for each monitor period by the sniffer process.

3.2 SMTP Traffic Data Collection

SMTP traffic data for analysis in this research does not involve the e-mail's content itself. The process of data collection only recorded the SMTP packets' TCP/IP header information that includes capturing time, source IP, destination IP, flag sets (SYN, FIN, ACK, RST), sizes of payloads, and so on. Summaries of what has been logged will be provided after every monitor process cycle is completed. SMTP traffic data was collected from two difference monitoring sources: one was a national wide ISP's local network, and the other was Loughborough University (i.e. the Loughborough University mail servers). A 24-hour period was set as the monitoring period for data collection as it relates to a standard human activity cycle. It will later be shown that the traffic characteristic for SMTP activity follows a 24-hour pattern.

In the national wide ISP's local network, there were known to be no legitimate email servers. Thus, all the hosts were legitimate email clients, spam relay hosts or illegitimate mail servers. Traffic data from this network is therefore helpful to indicate the SMTP traffic characteristics of legitimate email clients and spam relay hosts. The university email servers by contrast are legitimate servers with effective management. So the data from these servers can help to find the difference between a legitimate server's traffic and the spam relay's traffic by comparing with the data collected from the two networks.

3.2.1 Data from a National ISP's Network

We have access to a national ISP's traffic which had been previously collected by Dr Peter Sanford. The SMTP traffic data was collected from a national local network. Over 70 hours' SMTP traffic was logged and more than 10000 IP addresses were involved in this monitoring process. The SMTP packets' header information was recorded in detail.

The packets, which have a destination port value of 25 with the SYN flag set, were

logged. These packets are closely correlated to TCP connections with mail servers [25]. In section 3.1.2, it tells that a host firstly sends a packet with SYN flag set for requesting to establish a connection according to TCP three-way handshake protocol. And a host attempted to establish a TCP connection sent one and only packet with SYN flag set in both a completed connection and an uncompleted connection. SMTP packets are all have a port value of 25 according to the Simple Mail Transfer Protocol. Therefore, the number of packets with a SYN flag set and port value 25 is correlated to the number of SMTP connections requested to establish. An analysis tool named the SMTP Log Analyzer was used for analysis of the SMTP traffic connection profiles. SMTP Log Analyzer was written by a previous researcher in the HSN group at Loughborough University. It can display every 24-hour monitor period's distribution of packets for each source by loading the traffic data, and is also able to present all the Destination IP addresses related to a source appearing in a monitoring period. **Figure 3.3** shows the user's interface of the SMTP log analyzer.

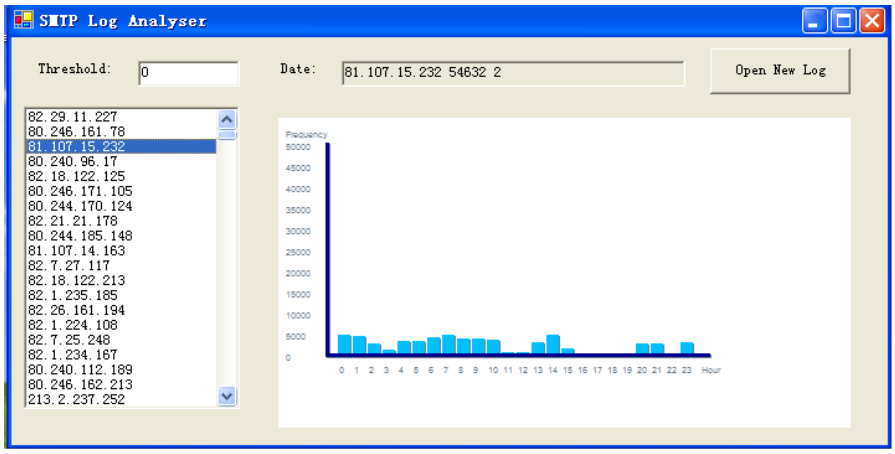

Figure 3.3: SMTP Log Analyzer

3.2.2 Data from the University Email Servers

The SMTP Traffic data from Loughborough University was collected from two email

servers in the university. These two email servers are legitimate email servers with effective management.

The data collected from one of the monitored university servers monitored represented all the SMTP traffic going in and out that server. A total of 15 days' traffic was gathered (fifteen 24-hour monitor periods), including 4 weekend periods. There were three days' data in which recorded all the SMTP packet header information including Capture time, Source IP, Payload Size, states of the flags in header (SYN, FIN, ACK, and RST) were record. The summaries, which come from the capture progress in these 3 days, counted the total number of SMTP packets, packets with SYN, packets with FIN, packets with RST, and the size of payload every 5 minutes. The other 12 days' data only included the summary of the number of SMTP packets, the number of packets with the SYN flag set, the number of packets with the FIN flag set, the number of packets with RST, and the size of payload transferred every 30 minutes.

The data from the other email server only recorded the outgoing SMTP traffic from that server. This monitoring recorded a total of 16-day's traffic including 4 weekends' periods. In these 16 days' traffic, 4 days' traffic was recorded with full details of each SMTP packet's header information and a summary was made every 5 minutes. For the other 12 days, a summary was made every 30 minutes and did not include the full detail about every SMTP packet header.

There were in total 14.6 Gigabytes of data collected from university mail servers in 31 days. A large amount of traffic data make the data processing and analysis difficultly and complicated. Therefore, there are in total 7 days' worth of SMTP traffic recorded with full detail of every SMTP packet's header information, and only summary data was collected in the other days. This summary data included the summary of the number of SMTP packets, the number of packets with the SYN flag set, the number of packets with the FIN flag set, the number of packets with RST, and the size of payload in each monitor time interval that is 30 minutes. The summary data is able to help to

analyze and understand the general and overall situation of legitimate email servers. At the same time the data with full detail is able to provide the particulars of SMTP traffic situation. Collection of summary data is not only able to reduce the total size of collection data and make the analysis processing of data easier, but also enough to provide valid data for the analysis with some data of full details together.

3.3 Summary

SMTP Traffic data was collected from different sources including legitimate email clients, legitimate email servers and spam relays in real networks (an ISP's local network and Loughborough University campus network). The header information of SMTP packets was logged by a sniffer. The information, which logged by the sniffer, included capturing time, source IP, destination IP, flag sets (SYN, FIN, ACK, RST), sizes of payloads, and so on. And email real content was never involved in the SMTP traffic data collection process.

SMTP traffic, which had been collected, will be analyzed in the next chapter to determine the differences regarding SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays.

Chapter 4: SMTP Traffic Characteristics of Legitimate emails clients, Legitimate Server and Spam Relay Hosts

The hosts that send SMTP packets can be divided into legitimated email clients, legitimate email servers and spam relay hosts. In this chapter, it will be found that legitimate email clients, legitimate email servers and spam relays have their own SMTP traffic characteristics. SMTP traffic characteristics of legitimate sites and illegitimate sites are different and may be used to distinguish each other in the network. Some methods based on analyzing SMTP traffic characteristics are suggested to identify spam relays in network at the last in this chapter.

The following sections review the related research work in SMTP traffic characteristics, followed by representing the differences regarding SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays as found by analyzing the SMTP traffic collected from real networks.

4.1 Related Work

A lot of email classification methods have been employed in anti-spam techniques. Most features used in these classification methods are able to be divided in two categories: one is per-email features, and the other is features calculated over a sending window.

Per-email features include the Single Email Multinomial-Valued Features (i.e. presence of HTML, presence of embedded image, presence of hyperlink, mime types of file attachment) and per email continuous features (i.e. number of attachments, number of words, and number of characters of subject or body). Features calculated over sending windows are number of senders, number of unique email recipients, ratio of email with attachment, and so on. [104]

Spam identification metrics have been introduced in papers in which characteristics of email servers, spam hosts (bots), and spam traffic are analyzed. These metrics involved size and type of attachment, number of recipients per email, email workload, email size, email inter-arrival time, path analysis, SMTP connection distribution, and so on [25][105][106][107][108][109][110][111][112][113][114]. Each metric can be a way to find out the spammers or spam traffic.

In the following sections, characteristics of legitimate email clients, legitimate email servers and spam relays will be determined with not only per email features but also features calculated over a monitor window by analyzing the SMTP traffic data collected from real networks. Difference in characteristics of the legitimate sites and spam relays will be presented in volume of SMTP connections, the rate of successful connections completed in SMTP connection requested, the number of similar emails, the rate of email response by the receiver, and SMTP connections distribution.

As previous section 1.1 said that spam emails are a part of “mass mailing”. Therefore one of significant characters of spam hosts is sending out a large volume of emails

(requesting to establish a large volume of SMTP connections). The volume of emails and SMTP connections are popular used parameters with other techniques for identifying spam hosts in the networks.

The legitimate mail traffic is two way traffic induced by social network [105], but the spam traffic is one way traffic [106]. 70% of spam emails are sending by spam bots. The spam-bot sends many spam mails, but it receives no mail because it doesn't have a domain which can be used for receiving emails and is only designed to send emails. Therefore, a high ratio of outgoing mails with a low ratio of email response by receivers is a characteristic of spam mail host [107]. This characteristic might be used with other anti-spam filtering techniques to improve the spam detection performance. This characteristic of legitimate and spam relays will be analyzed in the following sections in this thesis.

Email size also may be a parameter used with other filtering techniques to improving the effectiveness of spam identification. The sizes of non-spam emails are much more variable and have a much heavier tail in comparison with spam emails sizes [115]. Spammers typically send a large number of short e-mails. Most time they prefer to send large number of similar emails to mass receivers. The characteristics about numbers of similar email having similar email size from legitimate site and spam relays will be analyzed in this thesis.

Also analyzing the distribution of SMTP connections in a day is a popular way to identifying spam traffic. As a legitimate email server, the average number of requests exhibits large fluctuation over 24 hours [109]. The load is lightly in the early hours in a day. Then it gradually increases. It also exhibits self-similar behaviors [109]. Traditional non-spam e-mail traffic presents two distinct and roughly stable regions: a high load diurnal period (i.e., working hours), and a low load period covering the evening, night and early morning. On the other hand, the intensity of spam traffic is roughly insensitive to the time of the day [105]. As observed for daily load variations,

the impact of spam on the aggregate traffic is a less pronounced. SMTP connection characteristics were analyzed in [25]. It was found that the characteristics on distributions of SMTP connections can be used to detecting spam relays in the network. SMTP connection characteristics of legitimate email clients, legitimate email servers and spam relays were analyzed in this thesis.

In this thesis, a new characteristic on successful connection rate completed will be analyzed to find the difference between the legitimate email user and spam relays. Because the rejection of the connection requests from spam relays, it could be a way to be used with other anti-spam filtering techniques to improve the performance of the anti-spam filters

4.2 SMTP Traffic Characteristics of Legitimate Email Clients

In the national ISP's local network, there were no legitimate email servers. All the SMTP traffic data should therefore come from legitimate email clients. After the analysis of the data, we are able to suggest that there were also some spam relay hosts.

The ISP's local network data represented three days of SMTP traffic collected by a gatherer in a national ISP's network. There were a total of 2865 separate Source IP addresses in this data set. In other words, 2865 hosts' three days SMTP traffic was recorded in this data set. 2865 hosts were corresponding to these users who are in different careers, ages, background, habits, and so on. According to ISP's policy on this network using, there should be no illegitimate email servers and spam relays. Traffic data was analyzed, and it was found that most hosts in this network were legitimate email clients, also there are a few of illegitimate users (illegitimate email servers and spam relays). The following words in this section will try to determine SMTP characteristics of legitimate email clients by analyzing this SMTP traffic date set generated in a local network in which most hosts are legitimate clients. Significant characteristics of legitimate emails clients will be represented in this section. **Figure 4.1** shows the distribution of the number of Source Addresses, which are divided into different groups by the number of SMTP connections that they try to establish in a day.

Number of Connections	Number of Source IP	Source IP / Total logged Source Addresses
0~10	2260	78.61%
10~50	288	10.02%
50~100	59	2.05%
100~500	91	3.17%
500~1000	45	1.57%
1000~5000	51	1.77%
5000~10000	27	0.94%
10000~50000	49	1.70%
over 50000 ~	5	0.17%

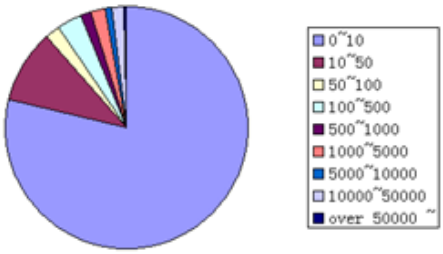

Figure 4.1: Distribution of Source IP by Number of Connection Established

Figure 4.1 shows that over 90% of hosts try to establish less than 100 connections with email servers in a day. This suggests that a legitimate email client may establish a few connections to mail servers each day. By analyzing the SMTP traffic data set, it was found that a legitimate email client would be usually expected to make limited connections in a few hours to one or several mail servers a day. For much of the day, it would remain silent. A 24-hour period is relates to a standard human activity cycle. Therefore, a 24-hour daily pattern in a particular day is able to more accurately represent the traffic characteristics for SMTP activity of a host. An averaged distribution over many days can shows the characteristics in a quite long period, but it also attenuates the variations in the distribution figures in a-day period. It is not expected to identify a spam relay by many days monitoring work, so we focus on analyzing characteristics for SMTP connections distribution in a particular day in this thesis. **Figure 4.2** shows the distribution of SMTP connections established by a typical legitimate email client in a particular day. A day's connection distribution pattern could be used to distinguish legitimate email client from legitimate email servers and spam relays which will be analyzed in the following sections.

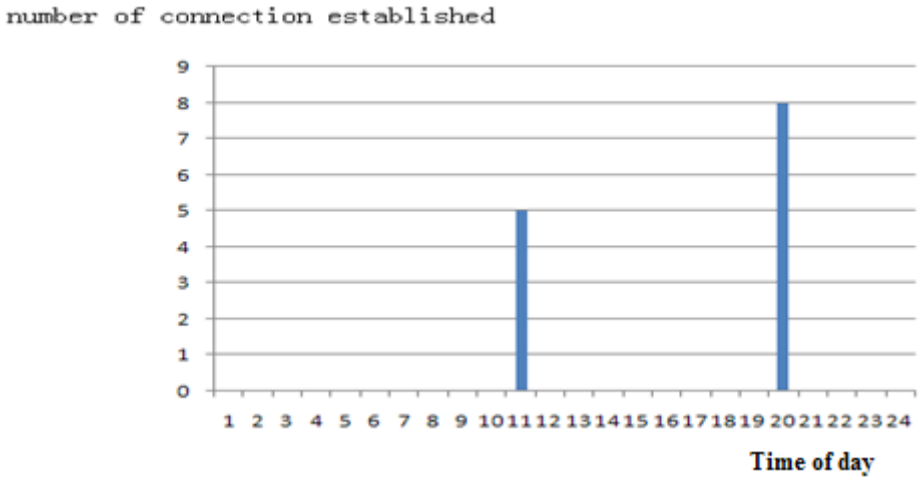

Figure 4.2: SMTP Connection Distribution of a Typical Legitimate Email Client

In Chapter 1, it has stated that spam emails are a part of a "mass mailing". Therefore most of the time a spam relay establish a lot of connections in a day. So in simple

term, it may be possible to separate legitimate email clients from spam relays in a network by counting the number of connections established in a monitoring period. Characteristics on volume of connections established by spam relays will be determined in later sections by analyzing SMTP traffic data collected from real network.

There may also be some illegitimate hosts sending limited number of emails to a target group of people for some selfish objective. However, this will affect the system. But this is a security problem, and it is not a problem of mass spam emails.

4.3 SMTP Traffic Characteristics of Legitimate Email Servers

The following subsections discuss the general characters of the University servers' SMTP traffic in terms of the volume of connections made, the ratio of the FIN/SYN flags, the payload-size of emails and the relation between patterns and time. This information could help to understand the SMTP traffic characteristics of legitimate email servers.

4.3.1 Volume of Connections

An obvious expected difference between email clients and mail servers is the volume of the connections. In the previous section 3.1.2, it has been explained that the number of packets with a SYN flag set from a host is closely correlated to the number of TCP connections which the host requested to establish. Therefore, the number of packets with SYN flag set was used to count the volume of connections that a monitor host requested to establish. A mail server makes a lot of connections every day, and a lot of emails are passed. There were nearly 20800 connections requested to establish by each Loughborough University email server on average every weekday, and 5020 connections on a weekend day. **Figure 4.3** shows the number of connections for thirteen days. The first ten bars are for the weekdays, and the last three are for the weekends.

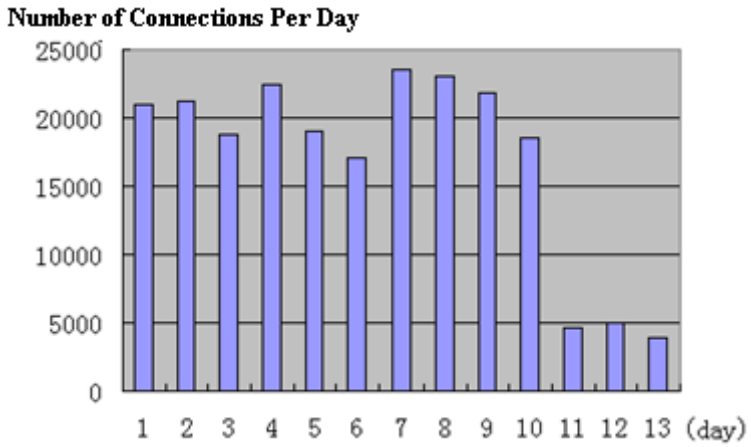

Figure 4.3: Number of Connections Requested to Establish by University Server per Day

In the busiest day there are 23520 connections. The minimum number of connections is at a weekend, and it is 3947. **Figure 4.4** shows the distribution for the number of connections requested to establish by the monitored university email server in a particular day. In a busy hour, the monitored server established about 2000 connections. There are only about 100 connections at mid-night, which is the quietest period for a server in a day.

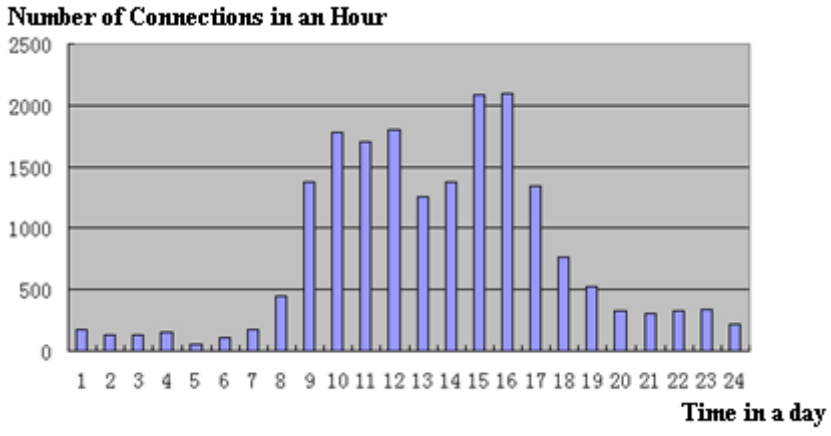

Figure 4.4：Distribution of Connections Established by University in a Particular Day

There are a lot of connections on the mail servers in every hour and every day. In contrast, an email client, as seen in the previous section 4.2, establishes only a few connections in an hour and only operates for a few hours in a day.

4.3.2 Ratio of FIN/SYN Flag Set

FIN/SYN flag set is defined as the ratio of the number of packets with the FIN flag set to the number of packets with the SYN flag set. The TCP 3-way handshake and tear down protocol tells us that there should be two packets with **SYN** and two packets with **FIN** sent by the Source IP for every completed connection. So for an ideal host without an unhealthy connection, the value of the FIN/SYN ratio should be 1.

From the data collected from the university servers, it is found that the average value of FIN/SYN is 0.82 in the traffic data collection period. The value during the weekday is higher than the value at the weekend. The average value of the FIN/SYN ratio is 0.57 at the weekends. Due to the uncompleted connections existed on each server, the value of FIN/SYN ratio is lower than 1. **Figure 4.5** shows the relationship between the value of FIN/SYN flag set and the number of the packets with a SYN flag set in every one-hour monitored time interval.

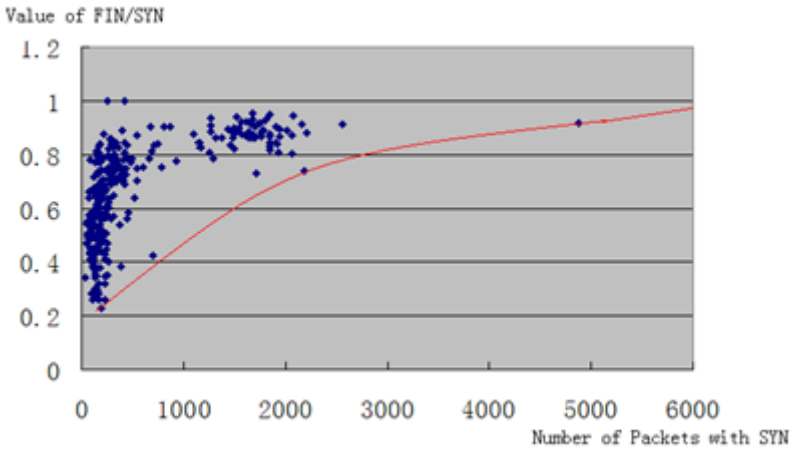

Figure 4.5: Distribution of the FIN/SYN ratio from a University Server

In contrast, most spam relay hosts may be expected to have many uncompleted connections with a lower value of FIN/SYN flag set because connections requested from these hosts could be refused by some legitimate servers and users for various reasons. For example a spam filter may stop unrecognized the email addresses which have been harvested from the web. **Figure 4.6** shows the distribution of the **FIN/SYN** flag set from a suspicious spam relay host in the ISP's local network. The packet transmission volume from this address is much higher than that expected from a legitimate email client. SMTP connections from this address were established in 6 hours in two time intervals, and each interval is 3 hours. The profiles of these connections in these two time intervals are the same. The profile of SMTP connections from this host is not expected from a legitimate mail server. In this case, it appears as expected a spam relay host sends spam emails cyclically and periodically.

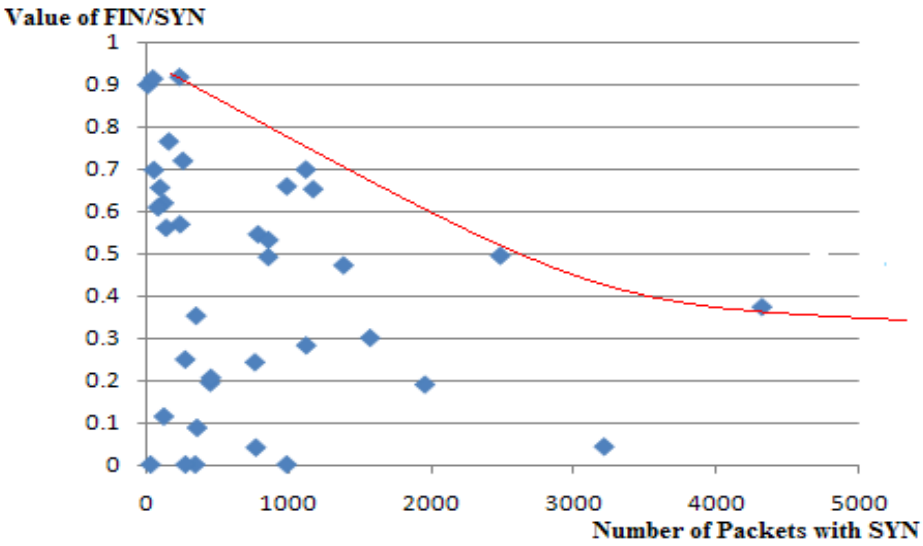

Figure 4.6: Distribution of the FIN/SYN ratios from a Suspicious Spam Relay

The different distributions in **Figure 4.5** and **Figure 4.6** point out that there may be a way to distinguish the legitimate email servers from spam relay hosts by the value of the FIN/SYN ratio when both of them generate a lot of SMTP connections.

4.3.3 Payload for the Emails on Servers

Content filtering is often used to fight spam emails by identifying what is in the emails. Actually a spammer sends similar emails to a lot of different email addresses in an outbreak most of the time. Similar emails should have similar size data contents. In other words, the payload for each connection should be similar for spam traffic. Some anti-spam techniques based on identifying similar contents and payloads in spam email have been developed. [116][117]

According to the TCP tear down protocol, there is a packet with a FIN flag set sent by the Source Address after the data transmission. Therefore the ratio of payload size to the number of packets with FIN flags set in a short time could be used to indicate if there are the connections with different payloads. **Figure 4.7** shows the values of Payload/FIN for SMTP traffic on a University server. The monitoring time interval is 5 minutes.

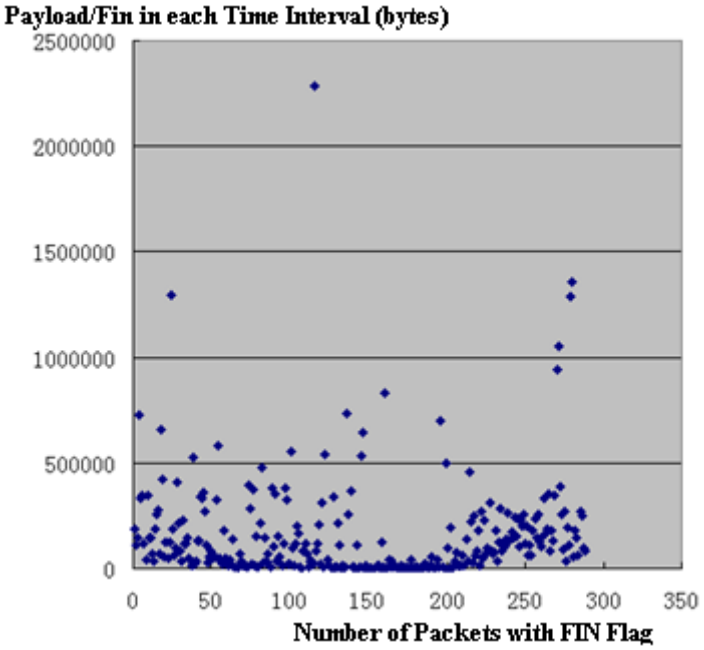

Figure 4.7: Distribution of Payload/FIN in Each Time Interval (bytes)

It is seen that many SMTP connections generated by this university server have different sizes of payload. In contrast, SMTP connections from a spam relay hosts sending similar emails should have similar size of payloads.

4.3.4 Patterns Related to Time

Human activity (study, work etc) is related to time. So a significant characteristic of the SMPT traffic from a legitimate mail server could be that the resulting profiles are related to time. **Figure 4.8** shows the average number of packets with a SYN flag set in every hour from one of the University Servers for 3 weekdays.

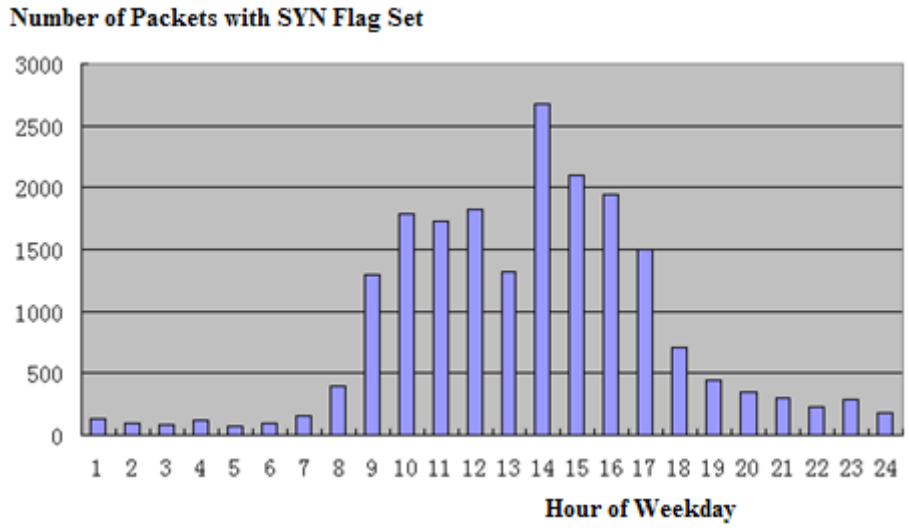

Figure 4.8: Distribution of Packets with SYN Flag from a University Server on Weekdays

Figure 4.4 has already shown a distribution of packets with SYN flag set from a university server for a 24-hour monitoring period on a weekday. The data comes from the same university server but on different days. The two profiles are visually very similar. From 0:00 AM to 7:00 AM, it is early morning, and in this period few people have got up to deal with emails. So this period is the quietest period in a day with an average connections number about 100. Then since 8:00 am, people have started to join work so that the number of connections is increasing. At 10 o'clock in the morning,

most of people have already been working and studying. So from 10:00 am to 12:00am, the number of packets with SYN flag is stable. There is a dramatic decrease around 13:00 pm not only in **Figure 4.4** but also in **Figure 4.8**. It is thought that most people's lunch break time is around 13:00 pm. After the lunch break, the number reaches to the peak of the day. It could be explained that the most popular thing is checking and dealing with the emails after people come back from the break. The number of connections has decreased since 15:00 PM, because people start to leave the campus. After 17:00 PM the number is less than 800, and then it keeps decreasing till midnight, which is the quietest point in a day. So SMTP traffic of a legitimate university email server is related to the time in the day.

Other days' profiles were also investigated, and the Kolmogorove-Smirnov Test was used to analyze these profiles. The Kolmogorove-Smirnov Test (K-S Test) is a non-parameter test for equality continuous, one-dimensional probability distributions. K-S test not only can be used to compare a sample to a reference probability distribution (one sample K-S test), but also can be used to compare to two samples (two sample K-S test) [118] [119]. The numbers of SMTP packets with SYN Flag set in every one-hour interval from three independent 24-hour monitor periods are shown in **Figure 4.9**.

Day 1: 2085, 2100, 1345, 763, 523, 323, 301, 328, 218, 178, 134, 127, 148, 50, 110, 174, , 340,444, 1372, 1775, 1702, 1800, 1252, 1381.

Day 2: 1848, 1672, 1137, 570, 424, 420, 297, 162, 246, 160, 127, 83, 74, 155, 91, 91, 148, 370, 1215, 1716, 1680, 1591, 1280, 1497.

Day 3: 2373, 2079, 2007, 777, 372, 320, 308, 205, 289, 175, 92, 67, 44, 56, 66, 102, 153, 380, 1295, 1873, 1811, 1524, 1442, 5152.

Figure 4.9: Distributions of SMTP Packets with SYN Flag Set from 3 Weekdays

The maximum number of SMTP packets with the SYN Flag set in a one-hour interval in these three 24 hour monitor periods is 5152, and the minimum number is 44. There

are 826 SMTP packets with SYN Flag set every one-hour interval on average in these periods. Also, in the working time of University (from 9:00 AM to 6:00 PM), the numbers of SMTP packets with SYN Flag set is bigger than 800, by contrast with the number is less than 800 in off-working time. **Figure 4.10** shows the distributions of one-hour intervals, which are divided into different groups by using the number of the SMTP packets with SYN Flag set.

Number of SMTP packets with SYN Flag set in a one –hour interval	Number of intervals in Day 1	Number of intervals in Day 2	Number of intervals in Day 3
0~800	15	15	15
800~1600	4	5	3
1600~2400	5	4	5
Over 2400	0	0	1

Figure 4.10: Distributions of One-hour Intervals

The Kolmogorove-Smirnov Test for Day1 and Day 2, gives the statistic D= 0.0417. The Kolmogorove-Smirnov Test for Day1 and Day 3, gives the statistic D= 0.0833. The Kolmogorove-Smirnov Test for Day2 and Day 3, gives the statistic D= 0.0833. So the maximum result value of the Kolmogorove-Smirnov Test is 0.0833. The number of elements in a sample is n=24. For n=24, α =0.05, the table value [120] is 0.32286. In our tests the maximum value of D is 0.0833, which is less than the table value. So the distributions of SMTP packets with SYN Flag set are similar on university server on weekdays. And all of these profiles are related to time in the same way.

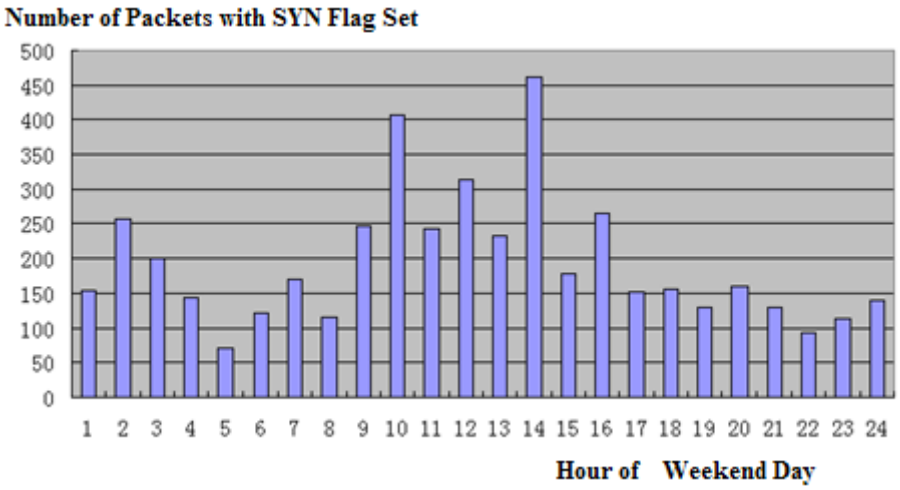

Figure 4.11: Distribution of Packets with SYN Flag Set from a University Server on a Weekend Day

It was found that the total number of packets with the SYN flag set at the weekend is much less than the number on a weekday. Also the profile is usually different to the profiles on a weekday. However the number of packets with SYN flag set in the daytime is greater than that for the number at night.

As such, patterns of SMTP traffic from a legitimate email server can be seen to be related to the time (different time in a day and workday or weekend), because these relate to a human's daily activities. However most spam emails from spam relay hosts are generated by automatic processes. Another section later in this chapter will point out that most SMTP traffic from spam relay hosts will be just related to the spammer's habits or the parameters set up for the Spam emails bots.

4.3.5 Ratio of Out/In SMTP Packets with SYN Flag Set

The incoming SMTP packets with SYN Flag Set are correlated to the connections

which the server is required to establish to receive the data. And the outgoing SMTP packets with SYN Flag set are correlated to the connections which the server tries to establish to send the data to others. As such, the ratio of Out/In SMTP Packets with SYN Flag set should be correlated to the response ratio, which is the rate representing the percentage of emails delivered by this server that are responded by the receiver. Few people will make a response to a spam email. Therefore, it may be possible to distinguish the legitimate email servers and emails spam relay hosts by the ratio of the Out/In SMTP packets with the SYN Flag set.

A total of 15 days SMTP traffic including the outgoing traffic and incoming traffic on a Loughborough University email server was analyzed. For a Loughborough University server, the average ratio of Out/In SMTP packets with the SYN flag set is 0.63. The minimum ratio is about 0.52 and the maximum is 0.87. There are of course suspicious spam relay hosts, which had sent a lot of SMTP packets with their SYN flag set, in the ISP's local network. However we haven't found any SMTP packets with the SYN flag set, in which the destination IP address is one of these suspicious IPs. Due to few people responding to a spam email, the ratio of Out/In SMTP packets with the SYN flag set of a spam relay should be a very large.

The difference between the legitimate email servers and suspicious spam relays is shown by the ratios of Out/In SMTP packets with the SYN flag set. Today, some legitimate email users send a lot of no-reply emails to their applicants. For example, a receipt will be send to you by email after you finish a shopping activity, but you will never make a response to this receipt via email. Therefore this ratio may be used to pick up suspicious spam relays in the network, but it is not good enough to identify spam relays.

4.4 SMTP Traffic Characteristics of Email Spam Relays

There are no legitimate email servers on the ISP's local network. Therefore if we remove the legitimate email clients' SMTP traffic, the rest of the traffic should come from suspicious spam hosts. Spammers always send a mass of similar spam emails to a lot of different email addresses [7]. Most email spam relays have unhealthy connection states. The following section presents some profiles from suspicious spam relay hosts' 24-hour SMTP traffic. From these profiles, the SMTP traffic characteristics of spam relays can be determined.

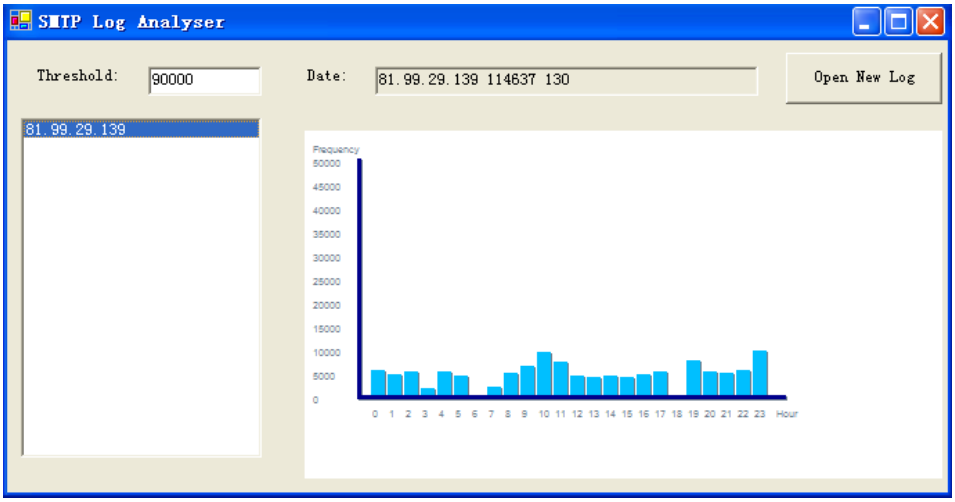

Figure 4.12: Profile of Suspicious Spam Relay (1)

Figure 4.12 shows a 24 hour profile of the connections for a host from the ISP's local network. In the day there were 20 one-hour monitored periods which have over 5000 connections. For a legitimate email server at Loughborough University, there are only about 2500 connections in the busiest hour in a day, and there are about 15 quiet one-hour time intervals, in which only about 100 connections are established each hour. Therefore the ISP's local network host sent out more emails not only in the daytime periods, but also at nighttime periods.

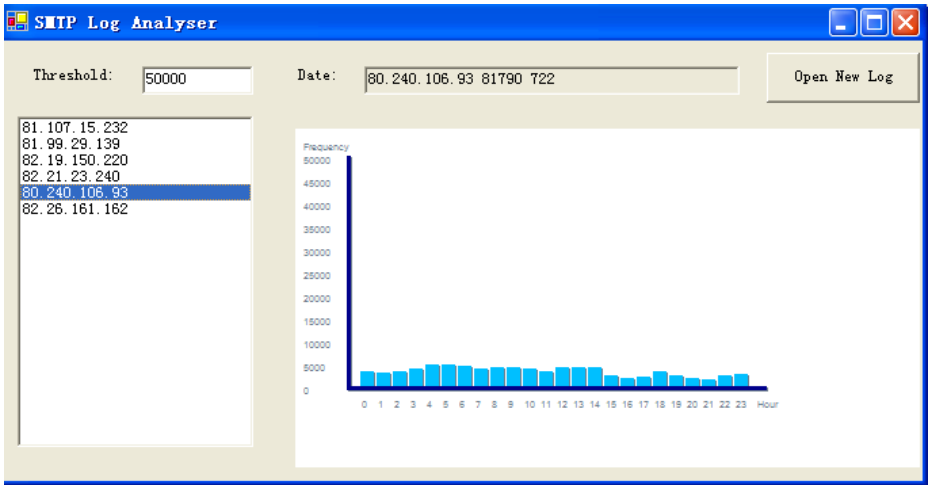

Figure 4.13: Profile of Suspicious Spam Relay (2)

The host with the profile shown in **Figure 4.13** has established more than 50000 thousand connections in 24 hours. Also there are over 2500 connections in every hour. Obviously it is not a legitimate email client, but also it could not be a legitimate email server. This is because the number of connections has been seen to be related to the time of day for legitimate servers and they do not keep sending a large number of emails every hour over the day. The profile shown by **Figure 4.14** shows the same situation of nearly constant SMTP traffic generation.

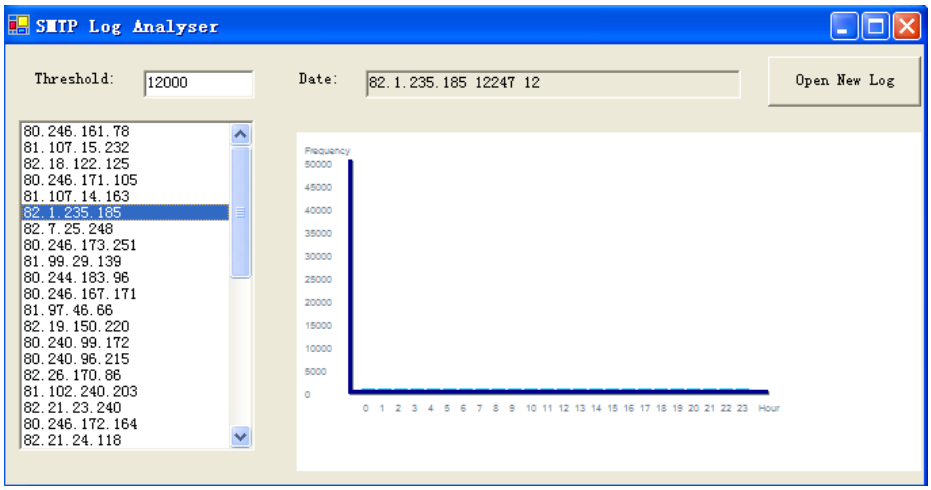

Figure 4.14: Profile of Suspicious Spam Relay (3)

In **Figure 4.14**, 12000 connections are sent over a 24-hour period. This is acceptable for

a legitimate server in a day, but the number of connections in every hour is almost exactly same. Hence, the traffic rate is not related to time, and this is not considered as a manual email transmission. In this case it could be said that a spam host or a spam relay machine was continuously sending out emails using automatic the spam emails transmission tools.

Both **Figure 4.13** and **Figure 4.14** show that the hosts send a lot of emails, irrespective of time of day, They could be the spam hosts or compromised hosts used as spam relay hosts.

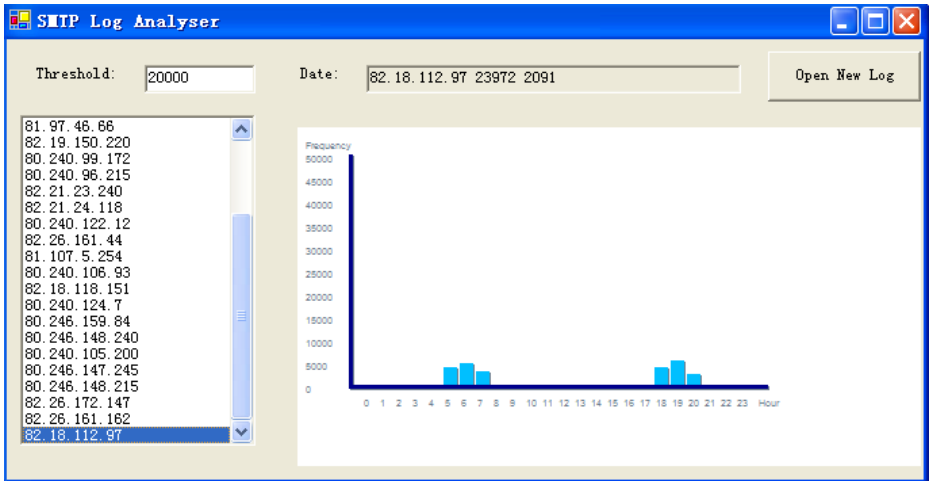

Figure 4.15: Profile of Suspicious Spam Relay (4)

Figure 4.15 shows that a host tried to establish about 30000 connections in 24 hour period. However, the 30000 connections were established in 6 hours in two time intervals, and each interval is 3 hours. The profiles of the connections in these two time intervals are the same. In this case, it looks as though a spam relay host sends spam emails cyclically and periodically.

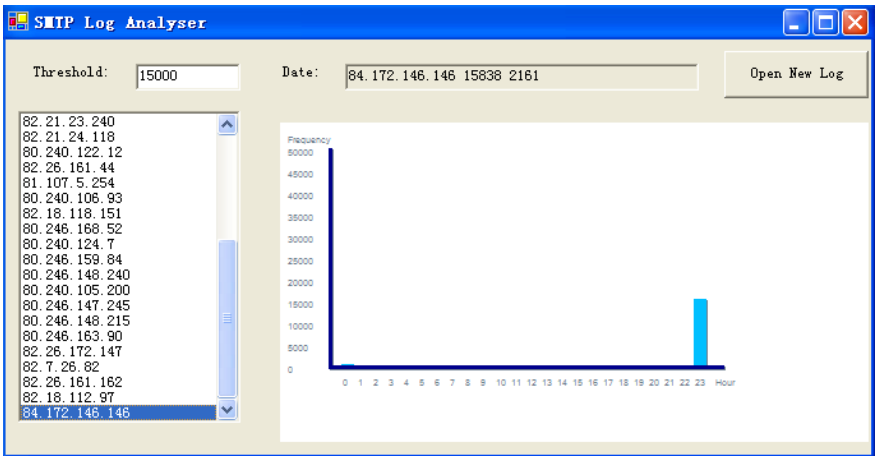

Figure 4.16: Profile of Suspicious Spam Relay (5)

In **Figure 4.16,** there are more than 15000 connections in a single one hour period, but nearly none in the other 23 hours during the day. Although this host exhibits quite different characteristics to the others previously shown, it is again immediately obvious that this profile does not fit that of a legitimate mail server or an email user client.

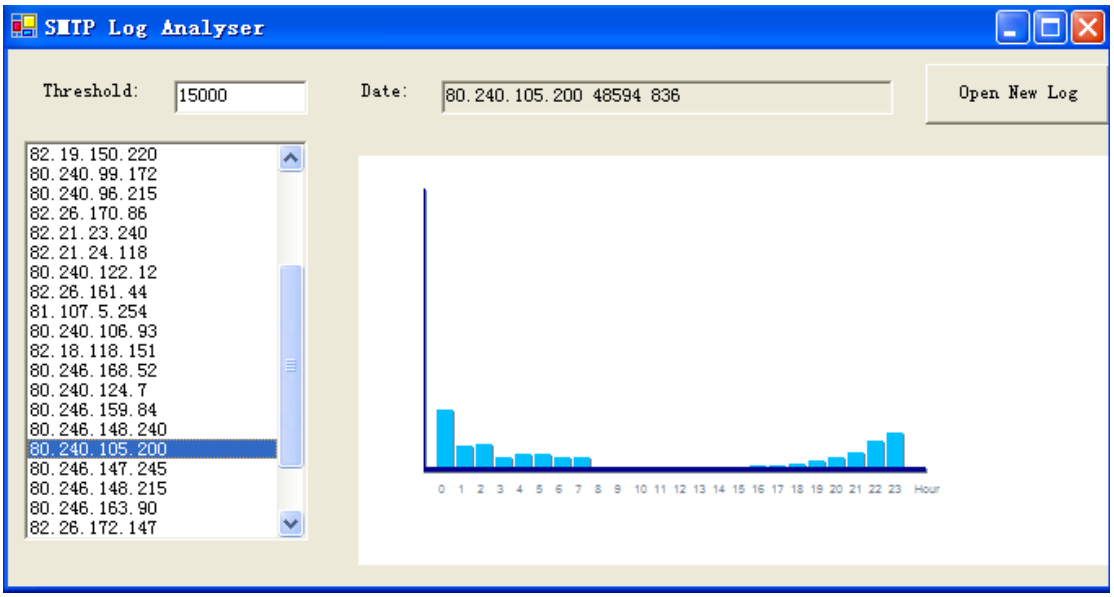

Figure 4.17: Profile of Suspicious Spam Relay (6)

The host with the profile in **Figure 4.17** establishes about 15000 connections in 24 hours. The profile is related to the time (similar to a legitimate server's). But in the

ISP's local network there are no legitimate email servers. So it could be an illegitimate email server in the network. We can't say it's a spammer, but it is against the policy of the ISP.

Overall, hosts sending spam emails have their own characteristic SMTP traffic. All of them establish a huge amount of connections, and most of the characteristic profiles are not related to the time of day as was seen to be the case for legitimate email servers.

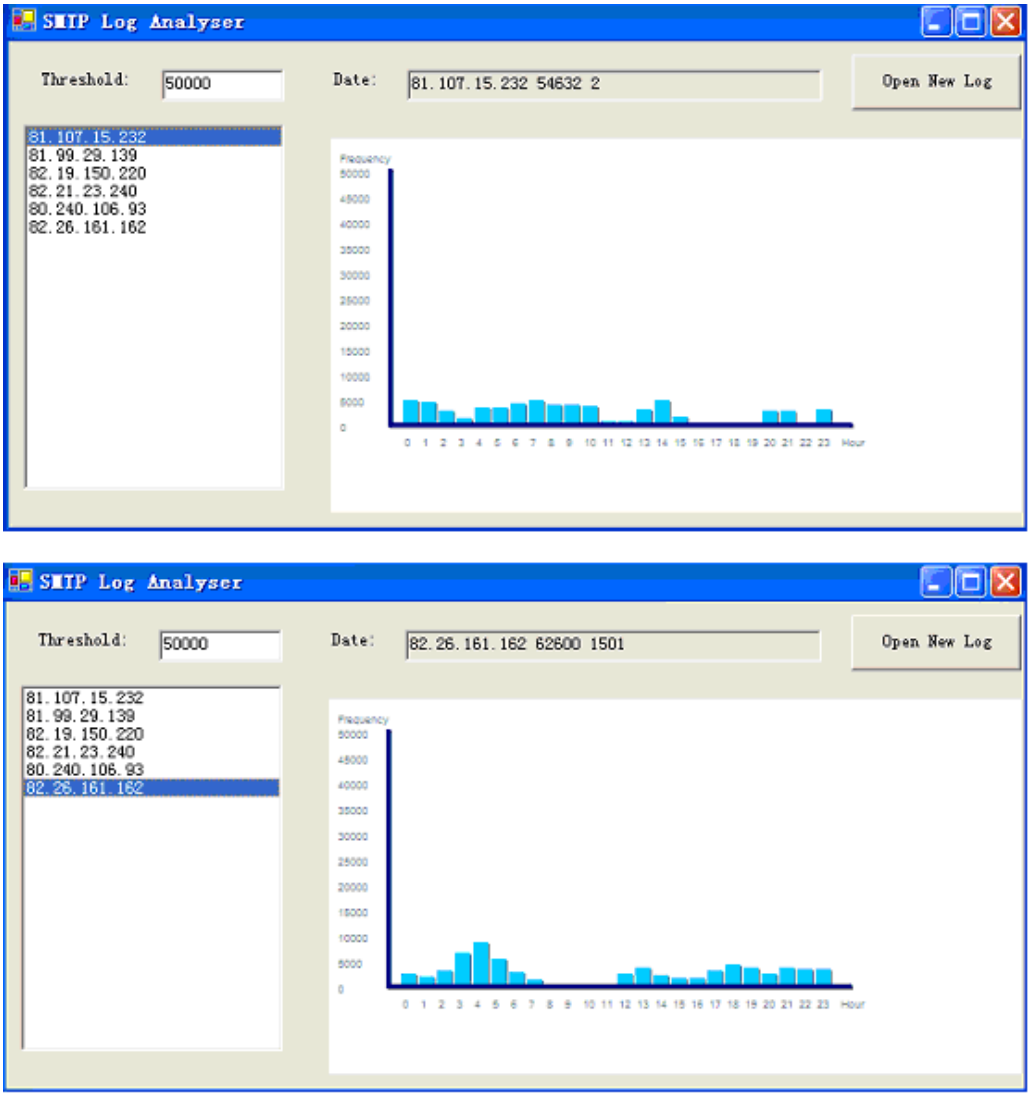

Figure 4.18: Profiles of Suspicious Spam Relays

There are another two profiles of connections from hosts in the ISP's local network, which are shown in **Figure 4.18**. Because there are no legitimate email servers inside

this network, these two profiles should have come from illegitimate mail servers or spam relay hosts. However these profiles do not indicate that they are from spam generators, and we must consider that some spam sources may not be distinguished from their traffic profiles.

The previous sections have described the SMTP traffic characteristics of legitimate user clients, legitimate email servers and spam relay hosts. All of these characteristics could be used to help in identifying the spam relay host in networks. The following section will provide an overview about the differences in the characteristics, which could be used to distinguish between the legitimate users (clients and servers) and spam relay hosts in the networks.

4.5 Differences in SMTP Traffic between Legitimate Users and Spam Relays

Email legitimate users in the network include legitimate email clients and legitimate email servers.

A legitimate email client is expected to establish a few connections over hours to one or several mail servers. Most of the time, it is silent. The SMTP traffic from a legitimate mail server is related to time of day. A lot of connections are established every day, but most of these connections are concentrated in a particular period that is related to human activity. Various emails coming from different writers are passed through a legitimate mail server. A legitimate server has been shown to have a stable number of emails passed through daily by the Kolmogorove-Smirvon Test analysis. Because of the responses of the email receivers, most of these servers should have not only outgoing SMTP traffic but also incoming SMTP traffic.

A mass of similar emails are delivered to a huge number of destination addresses by a spam relay host repeatedly. The profiles of SMTP traffic from a spam relay host do not relate to time in most cases. They are related to the spammer's working habits or the parameters set by the spam email-sending tools. Most spam emails are sent by automatic processes. And as such, cyclical and periodic phenomena often appear on the profiles of the Spam relay's traffic. Due to the use of some anti-spam techniques in the network, some of spam relay hosts could find that their attempts to establish connections were rejected. Also there is little incoming SMTP traffic for these spam relay hosts, simply because few people will respond the spam emails.

It should be possible to distinguish between email spam relay hosts and the legitimate email users by analyzing their SMTP traffic characteristics. The following subsections introduce some methods and parameters, which may be able to be used to identify spam relay hosts in the network.

4.5.1 Evaluation of the Successful Connection Rate verses the FIN/SYN Ratio

A low successful connection rate means a lot of connection requests from the host are refused. In other words, this host is not acceptable to most of the other hosts in the network. As such, it is a suspicious host with problems. Spammers harvest and download email addresses from the web, Usenets and the directories of email address used by the ISP as their spam emails' destination addresses. A lot of these email addresses could be invalid email addresses, which are unrecognized or nonexistent. Invalid destination email address could cause the rejection of the connection request. A lot of anti-spam tools have been applied by email users, for example Anti-Spam Assistant [121]. A spam host could have been detected by some of these and listed on the blacklists. Connection requests from these suspicious hosts would be rejected. So a host with a low successful connection rate has a big chance of being a spam host or a spam relay host. As the previous section has said, the ratio of the FIN/SYN flags set could be used to distinguish between the legitimate sites and illegitimate sites in networks when the hosts generated a lot of SMTP connections

4.5.2 Count the Total number of the Connections in a Particular Time Interval

Selfish Spammers always send out a large number of emails to reduce the price per email. A legitimate server has a stable number of emails passing through it daily. And email user clients are only expected to send several mails to one or several mail servers each hour. So the number of connections may not only help to identify the mail servers and spam hosts from the networks, but also distinguish spam hosts and legitimate mail servers. A particular time interval here refers to a monitoring period, for example an hour or a day.

4.5.3 Compare the Size of the Payload in Each Connection.

The size of the payload in a connection is related to the size of the real content in the related email. As the identified in the previous section 4.2.3, a spammer usually sends similar emails to a lot of destination addresses in an outbreak, but a legitimate email server generally passes different emails with the different payloads. In other words, the payload in each connection from a spam host in an outbreak should be similar most of the time. But for an email client and a legitimate server, different emails are sent with different payloads. Therefore, we may be able to identify the spam relay hosts by comparing sizes of payloads in connections in an outbreak.

4.5.4 Evaluate the Ratio of Out/In SMTP Packets with SYN Flag Set

Few people reply to spam emails. Most spam emails are deleted directly after they are confirmed as spam by receivers. So spam hosts and spam relay hosts always try to establish a lot of connections to send a lot of spam email, but there is little incoming SMTP traffic. This phenomenon could be used to help to identify the spam hosts in the network. But a white-list scheme is also necessary for detecting spammers if using this phenomenon, because not every legitimate email user expects the receiver give a response back. A white-list scheme can help to reduce the ratio of false positive errors.

4.5.5 Evaluate the Relationship between the SMTP Traffic, Time of Day and Human Habits (Human Actions)

Human actions are related to time of day and human habits. So SMTP traffic, which is created by people's email sending, should be related to time of day and the human's habits. As the previous sections have said, the profiles of legitimate email servers relate to time of day well, but most of profiles of spam relay hosts are not related to time of day.

As such, evaluating of the relationship between the SMTP traffic, Time of Day and Human Habits could help to identify spam relays, which have profiles different from human email profiles. Patterns of SMTP traffic could help identifying spam relays in this area.

These five methods never involve the actual email content. But they are correlated to successful connection rate (not-reject rate), volume of SMTP connections (number of emails sent out), payload size of emails in an outbreak, response rate (the rate of email response by the receiver), and the relationship to human activity (how humans generate emails). One of these methods maybe not enough on its own to identify spam relay hosts in the network, because of the false negative and false positive responses. It may be possible that combinations of these five methods in an anti-spam classifier could reduce such errors and improve the detection performance.

The following chapters design and evaluate an autonomous system, which combines these five methods to detect spam relay hosts in networks. Machine learning technology was also employed in this system. This approach detects spam relay hosts at the transmit stage via SMTP traffic characteristics, but it never involves the real actual email content.

4.6 Summary

SMTP traffic characteristics from legitimate email clients, legitimate email servers and spam relays are analyzed in this chapter. It has been shown that there are differences regarding the SMTP traffic characteristics of legitimate email clients, legitimate emails servers and spam relays. Also it was found that the SMTP traffic from the legitimate sites and illegitimate sites can be distinguished from each other by using SMTP traffic characteristics.

In this chapter, the understanding of the SMTP traffic characteristics from the different sources (legitimate and illegitimate sites) suggested some methods that might be used for detecting spam relays in the network. These methods will be employed in an autonomous spam relays detecting system, which will be introduced in the following chapter.

Chapter 5: An Autonomous System for Detecting Spam Relays by Using SMTP Traffic Characteristics

In this chapter, an autonomous spam relay detection system has been designed and presented. The components and the principles of operation of the system will be introduced in detail.

5.1 Components in the Autonomous System

In chapter 2, a typical pattern recognition system has been introduced in section 2.3. There are five parts in a typical pattern recognition system: sensor, segmentation, feature extraction, classification and post-processing. As an autonomous system for detecting spam relays by using SMTP traffic characteristics, our solution should achieve the following five functions: SMTP data collection, dealing with original traffic data to extract feature factors, launching the system decision scheme, classifying hosts in different categories and deciding on recommended actions to improve the system performance by outputs of the classifiers. Therefore in our autonomous spam relay detection system, we designed five elements corresponding to five functions: Sniffer, Pre-processor, Trigger, Classifiers and Post-Processor. **Figure 5.1** is the diagram of the proposed autonomous spam relays detection system.

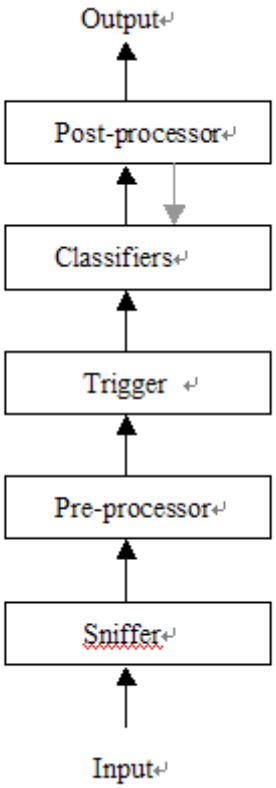

Figure 5.1: Diagram of Autonomous Spam Relay Detecting System

There are also two databases for helping the detection work in this system: SMTP Traffic Database and Spam Relay Database. The SMTP Traffic Database, in which all SMTP traffic information collected from the monitored network is stored, provides related traffic data to make the detection decisions. All the information about the spam relay hosts identified by the system in the monitored network is stored in the Spam Relay Database. Also the Spam Relay Database gives support to deciding on the recommended actions to improve the performance of the system. Both of these databases are automatically updated.

The following sections will introduce these five elements and two databases in detail.

5.2 The Five Parts of the Autonomous System

In our proposed spam relay detection system using SMTP traffic characteristics, there are five specifically designed elements: Sniffer, Pre-processor, Trigger, Classifiers and Post-processor. The following sections will present the functions of each element in more detail.

5.2.1 Sniffer

The sniffer element (also known as a network analyzer, protocol analyzer or packet analyzer) is computer software or hardware that can intercept and log traffic passing over a network or part of a network [122]. According to the particular types of networks in which a sniffer is used, sniffers can be divided into two categories: Ethernet sniffer or Wireless sniffer. As data streams flow across the network, the sniffer captures each packet and eventually decodes and analyzes its content according to the appropriate RFC or other specifications.

In Chapter 4, the SMTP traffic characteristics from legitimate sites and illegitimate sites were analyzed and compared. There, it was also recommended that some methods and parameters may be able to help to distinguish email spam relays from legitimate email users in networks. As our sniffer should provide SMTP traffic data that is necessary for detecting spam relays by system in the network, a special sniffer has been designed. The sniffer in this system collected the SMTP packets' TCP/IP header information from the network monitored as a sensor. This information forms the original inputs for the system.

The sniffer, which is employed in the proposed system in this thesis, captures the TCP/IP header information of each SMTP packet with the destination port value of 25. This original information captured by the sniffer includes Capturing Time, States of the Flags (SYN, FIN, ACK, and RST), and Size of the Payload for each packet, Source IP, and Destination IP. The SMTP traffic is monitored by the system, but the real content of

the email is never involved in this system.

5.2.2 Pre-Processor

The pre-processor deals with the entire SMTP traffic data logged by the sniffer. The SMTP Traffic Database is designed to store this useful header information that is necessary to make the detection decisions in our autonomous system. There is a data structure to store its information for each Source IP, which generates the SMTP traffic in the monitored network. **Figure 5.2** shows the data structure in the SMTP Traffic Database for an active host in the monitored network.

IP Address
Number of Packets with SYN Flag in 1^{st} Interval
.....
Number of Packets with SYN Flag in 24^{th} Interval
Number of Packets with FIN Flag in 1^{st} Interval
.....
Number of Packets with FIN Flag in 24^{th} Interval
Number of Outgoing Connection [7]
Number of Incoming Connection [7]
Payload-size of the Current Connection
Payload-size of the Last Connection
Number of Connection with Similar Payload-size [7]
Time of Last Packets Arriving

Figure 5.2: Data Structure of an Active Host in the SMTP Traffic Database

The pre-processor generates the SMTP Traffic Database, in which the Parameters are ready for the use of detection scheme. The pre-processor is written in the C programming language. When an arrival SMTP packet is logged by the sniffer, the pre-processor will check whether a data structure with the arrival packet's source IP is

available in the SMTP Traffic Database. If it is available, each parameter of this data structure will be updated according to the related TCP/IP header information of this packet. Otherwise, a new data structure will be generated to store the related TCP/IP information of this arrival packet. The following subsection will introduce the details about each parameter in the data structure and the updating process of these parameters in the SMTP Traffic Database.

- **IP Address（Source IP）**

The **IP Address** in this data structure is the Source IP of the SMTP packet. When the sniffer captures a SMTP packet, the source IP will be picked out firstly. Then it will be determined whether this source IP exists as **IP Address (Source IP)** of a data structure in SMTP Traffic Database. If this source IP is an inexistent **IP Address** in the database, a new data structure for this source IP will be created, and then the related parameters will be updated according to the arrival packet's TCP/IP head information logged by the sniffer. In this new data structure, the IP information in **IP Address** is the source IP. If this source IP is already available as an **IP Address (Source IP)** of a date structure in the SMTP Traffic Database, the related parameters will be updated in its exclusive data structure. The details about the updating process of each parameter will be represented in the following subsections.

Every valid active host, which has generated SMTP traffic captured by the sniffer, should have its own data structure in the SMTP Traffic Database. The system then sorts these SMTP packets in different categories by IP Addresses.

- **Number of Packets with SYN in N^{th} Interval**

Those packets, which have the SYN flag set and a destination port value of 25 (this indicates SMTP), are closely correlated to TCP connections with mail servers. The pre-processor counts the number of packets with the SYN flag set sent out by an active host in the network every monitoring time interval, then stores this number to the

related place（**Number of Packets with SYN in N^{th}Interval**）in the data structure which belongs to the active host. When time passes from an old interval to the following new interval, the number in the column related to the new interval will be set as 0. Then, when each new packet with the SYN flag set from the active host arrives, the number in the related column will increase by 1.

The monitoring interval of the system is set up as one hour. Therefore there are total 24 monitoring intervals in a day. Parameters in columns for **Number of Packets with SYN in N^{th} Interval** are only related to the most recent 24 hours. Any overtime parameters will be replaced by 0. The number in each time interval is correlated to the number of SMTP connections which the host tried to establish in this time interval.

The parameters of **Number of Packets with SYN in N^{th} Interval** not only present the distribution of SMTP connections in the most recent 24 hours, but also are necessary parameters for the calculation of the ratio of FIN / SYN flags set, which has been present in Chapter 4 as a possible way to distinguish spam relay hosts from legitimate email users in the network.

- **Number of Packets with FIN in N^{th} Interval**

The pre-processor not only counts the numbers of packets with the SYN flag set, but also counts the number of packets with the FIN flag set in each time interval. **Number of Packets with FIN in N^{th} interval** is the number of the packets with the FIN flag set in the N^{th} time interval. This number is correlated to the number of completed SMTP connections established by the active host in this time interval. These parameters are also necessary to calculate the ratio of the FIN/SYN flag set. The time interval of **Number of Packets with FIN in N^{th} interval** corresponds to the time interval of the **Number of Packets with SYN in N^{th} Interval**

When time turns into a new interval, the number in the column related to the new interval will be set as 0. Then, if a new packet with FIN flag set from the active host

arrives in this interval, the number in the related column will increase by 1. Parameters in columns of **Number of Packets with FIN in N^{th} Interval** are only related to the most recent 24 hours. Any overtime parameter will be replaced by 0.

- **Number of Outgoing Connection [7]**

Number of Outgoing Connection refers to the number of connections which an active host tried to establish. It is closely correlated to the number of SMTP packets with the SYN flag set, in which the source IP is the same as the value in **IP Address**. The pre-processor counts this number every day by calculating the sum of parameters in columns for **Number of Packets with SYN in N^{th} Interval**.

There are seven numbers stored in "**Number of Outgoing Connection [7]",** and each number responds to a day. Therefore it represents seven days' distribution of outgoing SMTP connections. The oldest number will be replaced by the new one that is generated after seven days. Parameters in Number of Outgoing Connections are not only able to present the total volume of SMTP connections attempted to be established by an active host in the recent seven days, but also they are essential parameters for calculating the ratio of Out/In SMTP connections.

- **Number of Incoming Connection [7]**

Number of Incoming Connection is referring to the number of connections which an active host are requested to establish a connection. It is correlated to the number of SMTP packets with the SYN flag set, in which the destination IP is same to the IP in **IP Address**. Parameters in **Number of Incoming Connection [7]** are also essential to the calculation of the ratio of Out/In SMTP connections**.**

There are seven numbers corresponding to the most recent seven days kept in **"Number of Incoming Connection [7]"**. Every arrival of a SMTP packet with SYN flag set, whose destination IP is the same as the value in **IP Address** in the structure,

will add 1 to the corresponding number in **Number of Incoming Connections [7]**. After system has been operational for seven days, the next arrival one will cause the value of 1 to take the place of the oldest value.

- **Payload-size of the Current Connection**

This part in the data structure is actual a counter, which is used to calculate the size of payload in each SMTP connection logged by the sniffer. The size of payload for each completed SMTP connection is related to the content of the corresponding email. Similar emails should have similar sizes of payload. Our spam relay detecting system checks whether the target emails are similar emails by using the size of the payload of each completed SMTP connection, and the real email content is never involved. Each completed SMTP connection may include several SMTP packets. **Payload-size of the Current Connection** in the data structure is the memory place to accumulate the sum of the payload sizes in related SMTP packets in a completed SMTP connection. The sum is the size of payload for the logged SMTP connection.

The method of counting the payload-size of the current connection is as follows:

For a completed SMTP connection, it will always start with a packet with the SYN flag set and finish with a packet with the FIN flag set. When a packet with the SYN flag set arrives, the parameter of **Payload-size of the Current Connection** in the corresponding data structure, in which the arrival packet's source IP is the same as the IP in **IP Address**, will be assigned as 0. Then the payload size of each packet comes from the same source IP will be added to the parameter, until a packet with the FIN flag set arrives. Then this parameter will be assigned to **Payload-size of the Last Connection**, which will be introduced in the following subsection. The parameter in **Payload-size of the Current Connection** will never be changed unless a new packet with the SYN flag set arrives from the same source IP. And the parameter will be never handed to the **Payload-size of the Last Connection** unless a new packet with the FIN

flag set arrives from the same source IP.

- **Payload-size of the Last Connection**

The parameter in **Payload-size of the Last Connection** is the size of payload in the most recent SMTP connection from the corresponding IP. In the previous subsection, it has already been explained that the parameter in **Payload-size of the Last Connection** would be replaced by the one in **Payload-size of the Current Connection** after a SMTP connection has completed. Before this replacement, these two parameters are compared to identify whether the payload-size of the current SMTP connection is similar to the last one, which was from the same source.

- **Number of Connections with Similar Payload-size**

When a SMTP connection has completed, the parameter in **Payload-size of the Current Connection** is the size of the payload for this connection. Then it will be compared with the parameter in **Payload-size of Last Connection**. If these two parameters are similar, the related parameter in **Number of Connections with Similar Payload- size** will plus 1. Then the parameter in **Payload-size of Last Connection** will be replaced by the parameter in **Payload-size of the Current Connection**. Most spammers send a large number of similar emails in an outbreak. So if the number of connections with similar payload-size is over the threshold of the detection system, the active host may be a spam relay host.

There are a total of seven numbers, which correspond to the most recent seven days, are recorded in **Number of Connections with Similar Payload-size**. After seven days, the next arrival one will use 1 to take the place of the oldest number.

- **Time of Last Packet Arriving**

The capturing time of the last SMTP packet from an active host is recorded in this part of the related data structure, in which the IP in IP Address is the same as the IP address

of the active host. This time is related to the host's last activity logged by the sniffer. According to this time, the update schemes of the system can remove the expired data in the database. For example, the data structure, in which an IP Address has kept silent over seven days, will be removed from the database. Update schemes will be detailed in another section later in this chapter.

5.2.3 Trigger

The Trigger element is designed to increase the operation efficiency of the detection system. A lot of IP addresses will be involved in the spam relay detection process by our system in monitored network. Also mass SMTP packets will be logged by system to make detections. It is not possible and necessary to pass the related information to classifier to make decision every single time when a SMTP packet is logged by the system. If we did, it will cost lots of resources and degrade the system performance. Therefore, only a portion of data from the suspicious spam relay hosts will start a trigger and be sent to the classifier for identification.

In Chapter 4, it was found that most hosts were the legitimate email clients in a real network. As a legitimate email client, it is expected to send several emails in hours every day. But the definition of spam emails discussed in Chapter 1 tells us that the spammer always send mass spam emails to a lot of receivers. A legitimate email server sends fewer emails than a spammer sends in a session most of the time. So in our system, the number of connections, which a host tried to established, is used to trigger the handover of the related data to the classifier for identification.

The number of SMTP packets with the SYN flag set is closely correlated to the SMTP connections established by hosts, so it is used to start the trigger in our system. There are two numbers which could start a trigger in our system: one is the number of SMTP packets with SYN flag set in current monitoring time interval ($\boldsymbol{T_{current}}$), which is the parameter in the current column for **Number of Packets with SYN in N^{th} Interval** in

the SMTP Traffic Database; and the other is the total number of SMTP packets with SYN flag set in recent 24 hours ($\boldsymbol{T_{total}}$), which is the sum of all parameters in **Number of Packets with SYN in $\boldsymbol{N^{th}}$ Interval** in the SMTP Traffic Database. In our detection system, a time interval is set as one hour. A day is divided into 24 time intervals.

The trigger is designed as a threshold based system. There are two types of thresholds for the trigger system: one is the average number of SMTP packets with the SYN flag set from hosts in current monitoring interval in the SMTP Traffic Database ($\boldsymbol{T_{current-th}[i]}$, the current time interval is the i^{th} interval in a monitor day, i=0,1,2,…,23.), which corresponds to $\boldsymbol{T_{current}}$, and the other is the average total number of SMTP packets with the SYN flag set from the hosts in the SMTP Traffic Database ($T_{total-th}$), which corresponds to $\boldsymbol{T_{total}}$.

$$\mathbf{T_{current-th}[i]} = \frac{\sum \textbf{Number of Packets with SYN in i}^{\textbf{th}} \textbf{ Interval}}{\textbf{Total Number of Hosts in SMTP Traffic Database}}$$

$$\mathbf{T_{total-th}} = \frac{\sum \sum_{\mathbf{i=1}}^{\mathbf{24}} \textbf{Number of Packets with SYN in i}^{\textbf{th}} \textbf{ Interval}}{\textbf{Total Number of Hosts in SMTP Traffic Database}}$$

In the i^{th} time interval only if $\boldsymbol{T_{current} \geq T_{current-th}[i]}$ or $\boldsymbol{T_{total} \geq T_{total-th}}$, the trigger will hand the related data to the classifier for spam relay identification. The update scheme of the system will calculate and update these thresholds every 24 hours in the update process. This update process will improve the performance of the system.

5.2.4 Classifier

When the trigger hands over the related SMTP data of a suspicious spam relay host, the classifier will identify whether this host is a spam relay host or not automatically. The classifier combines six algorithms. These algorithms correspond to the Ratio of FIN/SYN flag set, the Relationship between the traffic pattern and Time of day, Similar Payload-size Connections, Volume of Connections in Current Interval, Volume of Connections in the Recent 24 Hours and Ratio of Out/In SMTP packets with the SYN

flag set. These six algorithms have been presented as possible methods to identify spam relays in the network by using SMTP traffic characteristics discussed in the previous chapters.

Each algorithm will generate a value. According to these six values, a decision scheme based on weight factors will generate a single numerical value result. Then a final decision will be made by comparing the numerical value result with a threshold. The following sections will introduce the six algorithms and the decision scheme in classification.

5.2.4.1 Algorithms in the Classifier

- **Algorithm 1: Ratio of FIN/SYN Flag Set**

As Chapter 4 has said, the numbers of packets with the SYN flag set and the FIN flag set should be a pair in a completed SMTP connection. Therefore, this ratio is correlated to the percentage of the completed connections in all the connections attempted by the active host. When a host sends a lot of SMTP packets with the SYN flag set to try to establish connections, the ratio of FIN/SYN flag set could be used to distinguish between spam relays and legitimate email servers. A legitimate email client is only expected to establish several connections in each hours of a day, so a host (a legitimate email client) that established several SMTP connections will not be passed to classifier by trigger. Therefore the ratio of FIN/SYN flag set may be able to identify spam relay hosts by classifier in the network. An algorithm using this ratio was applied in our classifier.

Parameters in columns for **Number of Packets with FIN in N^{th} Interval** and **Number of Packets with SYN in N^{th} Interval** are able to be used to calculate the ratio of FIN/SYN flag set.

$$\text{Ratio of FIN/SYN flag set} = \frac{\sum_{i=1}^{24} \textbf{Number of Packets with FIN in } i^{th} \textbf{ Interval}}{\sum_{i=1}^{24} \textbf{Number of Packets with SYN in } i^{th} \textbf{ Interval}}$$

The coordinate of (FIN/SYN flag set, SYN), in which FIN/SYN flag set is the ratio of FIN/SYN flag set and SYN is the total number of the packets with SYN flag set, is introduced in this algorithm for identifying spam relays. In Chapter 4, when a host sends mass SMTP packets with SYN flags, the coordinates (FIN/SYN flag set, SYN) of legitimate email servers occupy the up-left area of the coordinate system. But the coordinates (FIN/SYN flag set, SYN) of suspicious spam relay hosts occupy the down-left area of the coordinate system. These coordinates are used to identify spam relay hosts in our system by using an average value approximation method. An approximation is a representation of something that is not exact, but still close enough to be useful. The following subsections will represent how to identify suspicious spam relays using this approximation method of Algorithm 1 in the system.

The system generates a group of coordinates of suspicious spam relay hosts from the Spam Relay Database. When a new coordinate (A, B) arrives, which needs to be identified, it will be put into this group. A is Ratio of FIN/SYN flag set, and B is number of SMTP packets with SYN flag set. In chapter 4, it has been presented that the location of this coordinate may be used to distinguish spam relays from legitimate email servers. The location depends on the values of A and B. It is also found that the value of A is related to the value of B. As a spam relay attempted to establish mass SMTP connections, the value of A could be decreased. So these coordinates will then be ordered by the number of SMTP packets with SYN flag set (the value of B). Now we have a group of coordinates (A_i ,B_i), where i = 1, 2…..N. Assume the coordinate, which needs to be identified, is (A_n ,B_n), where $n \neq 0$. In this group, we will use an approximation method to identify whether a host is a spam relay or not. Several coordinates in this group will be picked out for working out the identification threshold for this algorithm. These selected coordinates called nearest coordinates, because the values of B_i in these coordinates are closer to the value of B_n than the other

coordinates in this group. Then we use the average value of the values of A_i in these nearest coordinates as the threshold ($\boldsymbol{TH_{F-S}}$). If $A_n \leq \boldsymbol{TH_{F-S}}$, the host will be identified as a suspicious spam relay. There are a total of up to 2m values of A_i which are used to calculate the threshold ($\boldsymbol{TH_{F-S}}$) . There are m values of A_i from the m closest left side coordinates to the target coordinate (A_n ,B_n), and the other m values of A_i from the m closest right side coordinates to the target coordinate (A_n ,B_n). If there are not enough coordinates on any side of the target coordinate (A_n ,B_n) for calculating the threshold, the number of coordinates on the shorter side will be defined as the temporary m_t . In this case, m_t values of A_i from the m_t closest left side coordinates to the target coordinate (A_n ,B_n) and m_t values of A_i from the m_t closest right side coordinates to the target coordinate (A_n ,B_n) are used to calculate the threshold. If the target coordinate (A_n ,B_n) is the first one in the ordered group of coordinates, the m or m_t closest right side coordinates are used to calculate the threshold. If the target coordinate (A_n ,B_n) is the last one in the ordered group of coordinates, the m or m_t closest left side coordinates are used to calculate the threshold. The following words represent how the threshold ($\boldsymbol{TH_{F-S}}$) is calculated by using this group of coordinates for (A_n ,B_n). In our system m is set as 5.

If n=1:

While N≤ m; $TH_{F-S} = \frac{\sum_{i=2}^{N} A_n}{N-1}$.

While N>m; $TH_{F-S} = \frac{\sum_{i=2}^{m+1} A_n}{m}$.

If 1<n≤N-n:

While m<n; $TH_{F-S} = \frac{\sum_{i=1}^{m} A_{n-i}+A_{n+i}}{2m}$.

While 1<n≤m; $TH_{F-S} = \frac{\sum_{i=1}^{n-1} A_i+A_{n+i}}{2(n-1)}$.

If N > n>N-n:

While n<N-m; $TH_{F-S} = \frac{\sum_{i=1}^{m} A_{n-i}+A_{n+i}}{2m}$.

While N>n≥N-m; $TH_{F-S} = \frac{\sum_{i=1}^{N-n} A_{n-i}+A_{n+i}}{2(N-n)}$.

If n=N:

While N≤m; $TH_{F-S} = \frac{\sum_{i=1}^{N-1} A_n}{N-1}$.

While N>m; $TH_{F-S} = \frac{\sum_{N-m}^{N-1} A_n}{m}$.

- **Algorithm 2: Relationship to Time of Day**

The number of connections established by legitimate email servers relates well to time of day. But for a spam relay host, the connections number may be great in the network quiet period, or connections occur periodically and cyclically. In Chapter 4, it was found that a legitimate email server establishes over 90% of its SMTP connections in 8-hour period (8:00~ 16:00), which is the busy period every day. And the other two 8-hour periods (0:00~8:00 and 16:00~24:00) are very quiet. But the working period of a spam relay only depends on the spammer's fancy.

A simple method has been designed to identify spam relay hosts by using this phenomenon. Time is divided into three periods including two quiet periods (0:00~8:00 and 16:00~24:00) and one busy period (8:00 ~ 16:00) in a day. The ratio of the number of packets with SYN flag set in quiet periods / total number of packets with SYN flag set in a day is calculated. If the ratio is greater than the corresponding threshold ($\boldsymbol{TH_{Time}}$), the host will be considered to be a suspicious spam relay. The threshold ($\boldsymbol{TH_{Time}}$) will be generated by the system every 24 hours, which will be detailed in a subsection later in this chapter.

- **Algorithm 3: Similar Payload-size Connections**

For each active host's data structure in the SMTP Traffic Database, there are parameters in the columns for **Number of Connections with Similar Payload-size [7]**, which present how many similar payload-size connections have been established. Most spammers send similar emails in a session. If the number of similar payload-size connections from a host in the current day is over the corresponding threshold (**$TH_{payload}$**), this host will be identified as a spam relays.

- **Algorithms 4&5: Volume of Connections in Current Interval & Volume of Connections in Recent 24 Hours**

Both of these two algorithms are used to identify spam email sessions according to the spam relay host's characteristics of mass connections established. When the trigger hands a host to the classifier, the number of packets with the SYN flag set in the current interval from that host is greater than the corresponding average number in the network, or the number of connections in most recent 24-hour period from that host is greater than its corresponding average number.

There are another two thresholds (**$TH_{Vol-Current}$** & **$TH_{Vol-total}$**) from the Spam Relay Database, which relate to these two algorithms. If a host's volume is greater than the corresponding threshold (**$TH_{Vol-Current}$ or $TH_{Vol-total}$**), this host will be identified as a suspicious spam relay.

- **Algorithm 6: Ratio of Out/In SMTP Connections**

The ratio of Out/In SMTP connections is correlated to the reply rate to emails sent by the specific host. A legitimate email user will be replied after its legitimate emails are received by contact persons most of the time. But as a spammer, few people will respond to the spam emails sent by them. Therefore a host could be healthy if the amounts of incoming SMTP connections and outgoing SMTP connections have a

reasonable relationship. Otherwise, the host may be a suspicious spam relay. In Chapter 4, it has been said the ratio of Out/In SMTP Connections may be used to distinguish between legitimate users and spam relay hosts.

The number of SMTP Connections is closely correlated to the number of SMTP Packets with the SYN flag set. Therefore the ratio of Out/In SMTP connections is correlated to the ratio of Out/In SMTP packets with SYN flag set. The following describes how the system calculates an active host's ratio of Out/In SMTP Connections by using the data in the SMTP Traffic Database.

For an active host in the network, there is a data structure to store the related information in the SMTP Traffic Database. In the host's own data structure, there are seven numbers in the column for **Number of Outgoing Connections [7]**, and there are also seven numbers in the column for **Number of Incoming Connections [7]**.

While the sum of incoming connections is not 0:

The ratio of Out/In SMTP Connections ($\frac{\boldsymbol{OUT}}{\boldsymbol{IN}}\boldsymbol{Ratio}$):

$$\frac{OUT}{IN}Ratio = \frac{\sum_{i=1}^{7} \text{Number of Outgoing Connection [i]}}{\sum_{i=1}^{7} \text{Number of Incomming Connection [i]}}$$

If $\frac{OUT}{IN}Ratio >$ corresponding threshold($\boldsymbol{TH_{out-in}}$) in the system, this host will be considered as a suspicious spam relay host.

While the sum of incoming connections is 0:

The host will also be thought as a suspicious spam relay host.

5.2.4.2 Final Decision Scheme

The six algorithms will generate six results ($\mathbf{R_i}$), where i =1, 2……6. If the $\mathbf{i^{th}}$ algorithm identified the host as a suspicious spam relay, then $\mathbf{R_i = 1}$; otherwise $\mathbf{R_i = 0}$. The final job of classification is to make a final decision to identify whether the suspicious host is a spam relay host or not in the network. The following describes how this final decision is made by the system.

Final Decision Scheme Based on Weight Factors

Weight factors are estimated values that indicate the relative importance [123]. A weight factor is assigned to a variable to emphasize its contribution to a final effect or result. Most time a decision scheme based on weight factors is able to provide a more accurate result than the scheme, in which each variable is considered to contribute equally to the final result. In our system, weight factors are used to present the contributions of the six algorithms to the spam relay detection. There are a total of six weight factors ($\mathbf{W_i}$) corresponding to the six algorithms in the classifier.

Bayes' theorem is popularly used to indicate the contribution of an element to a final result. In Chapter 2, it was said that the Bayesian Content Spam Filter used this theorem to evaluate the contribution of each word in emails. The formula is $P(S|W) = \frac{P(W|S)Pr(S)}{P(W|S)Pr(S)+P(W|H)Pr(H)}$. if the overall probability that any given message is spam and the overall probability that any given message is not spam are the same, then $Pr(S) = Pr(H) = 50\%$. Then the formula $P(S|W) = \frac{P(W|S)}{P(W|S)+P(W|H)}$ is derived from that. In our system, this formula is used to evaluate the contribution of each algorithm to the final result.

The weight factor for each algorithm is:

$$\boldsymbol{W_i} = \frac{P(A|S)[i]}{P(A|S)[i]+P(A|H)[i]} \quad , \qquad i=1, 2, \ldots..6.$$

Where:

$\mathbf{P(A|S)[i]}$ is the probability of a host identified as a suspicious spam relay by the i^{th} algorithm being identified as a spam relay by the final decision scheme.

$\mathbf{P(A|H)[i]}$ is the probability of a host identified as a suspicious spam relay by the i^{th} algorithm being identified as a legitimate user by the final decision scheme.

Method for calculating $P(A|S)[i]$ and $P(A|H)[i]$:

When the i^{th} algorithm identifies a host as a suspicious spam relay, a counter will be launched .There are two numbers in this counter: one is the number ($\boldsymbol{N_s[i]}$) of hosts that have been identified as a suspicious spam relay by both the i^{th} algorithm and the final decision scheme, and the other is the number $\boldsymbol{N_l[i]}$ of legitimate hosts that have been identified as a suspicious spam relay by the i^{th} algorithm. Then the probabilities could be calculated

$$P(A|S)[i] = \frac{\boldsymbol{N_s[i]}}{\boldsymbol{N_l[i] + N_s[i]}}$$

$$P(A|H)[i] = \frac{\boldsymbol{N_l[i]}}{\boldsymbol{N_l[i] + N_s[i]}}$$

The final decision is: $\mathbf{D} = \sum_{\mathbf{i=1}}^{\mathbf{6}} \mathbf{W_i} \times \mathbf{R_i}$

The threshold for the final decision scheme of the system ($\mathbf{D_{threshold}}$) and weight factors ($\mathbf{W_i}$) are generated automatically by the Post-processor, which will be introduced in the following section. $\mathbf{R_i}$=1, when the $\mathbf{i^{th}}$ algorithm identified the host as a suspicious spam relay, and $\mathbf{R_i}$=0, when the $\mathbf{i^{th}}$ algorithm identified the host as a legitimate user.

If $\mathbf{D} \geq \mathbf{D_{threshold}}$, the host is a spam relay host in the network, otherwise it is a legitimate user.

5.2.5 Post-Processor

The Post-processor uses the output of the classifier to decide on the recommended actions. It is very important to improve the performance of the system. There are three main functions of the Post-processor in our system: generate the Spam Relay Database, generate the system parameters and thresholds, which will be used by the trigger, the classifier, and so on, and update the databases in the system. The following section explains how the Post-processor achieves these functions.

5.2.5.1 Generating the Spam Relay Database

There are two Databases to store and manage the hosts' information. The Pre-processor generated the SMTP Traffic Database, and the Spam Relay Database is generated by the Post-processor. This Spam Relay Database stores and manages valid information about the hosts that have been identified as spam relay hosts by the system.

After a host is identified as spam relay host by the classifier, the information of this host would be passed to the Post-processor. The Post-processor then generates the Spam Relay Database by processing this information. In Spam Relay Database each spam relay host has a data structure to record the related information. Then the information in the Spam Relay Database is used to generate a series of system parameters and thresholds to improve the performance of the spam relay detection process. This database expresses the characteristics of the identified spam relays.

The **Figure 5.3** shows a data structure of a spam relay host in the Spam Relay Database.

IP Address
Number of Packets with SYN Flag in 1^{st} Interval
.....
Number of Packets with SYN Flag in 24^{th} Interval
Number of Packets with FIN Flag in 1^{st} Interval
.....
Number of Packets with FIN Flag in 24^{th} Interval
Number of Outgoing Connection [7]
Number of Incoming Connection [7]
Payload-size of the Current Connection
Payload-size of the Last Connection
Number of Connection with Similar Payload-size [7]
Time of Last Packets Arriving
Ratio of OUT/IN SMTP Connection
Decision Results of Each Algorithms [6]
Results from Final Decision Scheme: D

Figure 5.3: Data Structure of a Spam Relay Host in Spam Relay Database

It is seen that the first 13 columns of the data structure of the SPAM Relay Database are the same as the data structure in the SMTP Traffic Database. When a host has been identified as a spam relay, the parameters in these 13 columns will be assigned to the corresponding columns in the Spam Relay Database. There are three additional columns in the data structure of Spam Relay Database: Ratio of OUT/IN SMTP Connection, Decision Results of each Algorithm, and Result from Final Decision Scheme. The details about the 3 additional columns of the data structure are explained in the following sections.

- **Ratio of Out/In SMTP Connection**

Algorithm 6 of the classifier calculates the ratio of Out/In SMTP connections, when the

host data was handed to the classifier by the trigger. Details about the calculation are presented in Section 5.2.4.1. After the host was identified as a spam relay host by the final classification, the value of this ratio would be assigned as a parameter in the column for **Ratio of Out/In SMTP Connection** of the data structure in the Spam Relay Database. If the number of incoming connections is 0, the ratio, whatever it is in this column, will be no meaning.

- **Decision Results of Each Algorithm [6]**

As the previous section 5.2.4.2 has said, each algorithm makes an independent decision of suspicious spam relay or not. If a host was identified as a suspicious spam relay host by an algorithm, a result valued "1" would be generated; otherwise, a result valued "0" would be generated. Therefore there are total six results valued "1" or "0" generated by the six algorithms in the classifier. These six values will be assigned in an area of **Decision Results of Each Algorithm [6]**.

Values in **Decision Results of Each Algorithm [6]** in the Spam Relay Database are used to calculate the Weight Factors of Algorithm. Weight factors present the contribution of the corresponding results from the algorithms to the final decision.

- **Result from Final Decision Scheme: D**

In section 5.2.4.2, Final Decision Scheme, it has been said that the final decision scheme will generate a result **D**. The value of **D** will be stored in the column for **Result from Final Decision Scheme**.

Also in that section, it was said that the final decision was made by comparing the value of **D** and the value of $\mathbf{D_{threshold}}$, which is the threshold for the Final Decision Scheme of the system. The value of $\mathbf{D_{threshold}}$ is calculated from all the values in columns for **Result from Final Decision Scheme** in Spam Relay Database.

So far, it can be said that the Spam Relay Database has two main functions: the one is to store and manage the spam relay hosts' information, which could help to understand spam activity and characteristics of spam relays, and the other is to provide the data to the Post-processor to automatically generate a series of parameters and thresholds, which are used by the system and improve the performance.

5.2.5.2 Generating Parameters and Thresholds

Another function of the Post-processor is generating a series of parameters and thresholds to keep the system operating automatically and to improve its performance. Information in the Spam Relay Database would be used to generate these useful values. The following subsections will introduce how the Post-processor generates these parameters and thresholds automatically in the system.

- **Generating the Thresholds for the Trigger System**

There are two thresholds used in the trigger system: one is the average total number of SMTP packets with the SYN flag set from the hosts in the monitored network ($\boldsymbol{T_{total-th}}$), and the other is the average number of SMTP packets with the SYN flag set from hosts in the current monitoring interval in the monitored network ($\mathbf{T_{current-th}[i]}$). The formulas for calculating these two thresholds have been presented in the previous section 5.2.3. And every 24 hours, new thresholds will be generated by using those two formulas.

- **Generating the Coordinates of (FIN/SYN flag set, SYN)**

In Section 5.2.4.1, it was said that **Algorithm 1 of the Ratio of FIN/SYN Flag Set** used a group of coordinates (FIN/SYN flag set, SYN) from the Spam Relay Database to identify whether a new arrival host is a suspicious spam relay. This group of coordinates is generated by the Post-processor by using data in the Spam Relay Database. Each coordinate corresponds to a data structure in the Spam Relay

Database.

The following formulas are used to calculate this group of coordinates (FIN/SYN flag set, SYN).

$$\text{FIN/SYN flag set} = \frac{\sum_{i=1}^{24} \textbf{Number of Packets with FIN in } \boldsymbol{i^{th}} \textbf{ Interval}}{\sum_{i=1}^{24} \textbf{Number of Packets with SYN in } \boldsymbol{i^{th}} \textbf{ Interval}}$$

$$\text{SYN} = \sum_{\text{i=1}}^{24} \text{Number of Packets with SYN in i}^{\text{th}} \text{ Interval}$$

Where:

Number of Packets with Fin in i^{th} Interval is the parameter in the column for **Number of Packets with Fin in i^{th} Interval** in the data structure in the Spam Relay Database.

Number of Packets with SYN in i^{th} Interval is that the parameter in the column for **Number of Packets with SYN in i^{th} Interval** in the data structure in Spam Relay Database.

By using **Algorithm 1**, a new data entry corresponding to the new arrival host will be generated to identify whether it is from a suspicious spam relay host.

When a new arrival host is identified as a spam relay by the Classifier, its coordinate of (FIN/SYN flag set, SYN) will be used for the next identification.

- **Generating the Threshold (TH_{Time}) for Algorithm 2 in Classifier**

Algorithm 2 identifies suspicious spam relay hosts by evaluating

$$\textbf{ratio} = \frac{\textbf{Total Number of packets with the SYN flag set in quiet periods}}{\textbf{Total number of packets number with SYN flag set in a day}}$$

We use the following formula to calculate the corresponding ratio of each spam relay host, which has been identified as a spam relay, by using the data structure in the Spam

Relay Database.

$$\text{ratio} = \frac{\sum_{i=1}^{8} \text{number of Packets with SYN in i}^{\text{th}} \text{ Interval} + \sum_{i=17}^{24} \text{number of Packets with SYN in i}^{\text{th}} \text{ Interval})}{\sum_{i=1}^{24} \text{number of Packets with SYN in i}^{\text{th}} \text{ Interval}}$$

The 95th percentile value of the ratios is picked up as the threshold ($\boldsymbol{TH_{Time}}$) for **Algorithm 2**. Every 24 hours, a new threshold will be generated

Percentile Value for Thresholds:

When the system generated the threshold for **Algorithm 2**, the 95th percentile was used to help generate this threshold. The standard definition of a reference range for a particular measurement is defined as the prediction interval between which 95% of values of a control group fall into, in such a way that a total of 5% of sample values will be less than the lower limit or larger than the upper limit of the interval [124]. The value of 95% is therefore used as the percentile value for thresholds in this thesis. This value could be changed by network administrators to other values to meet the detection requirements before the system is applied in a real network. This percentile value for thresholds determines the percentage of hosts in Spam Relay Database that are above to the threshold of the system. Therefore this value affects the performance of system. A series of tests have been conducted to determine the effect of changing this percentile value on false positive rates and false negative rates.

- **Generating the Threshold ($\boldsymbol{TH_{payload}}$) for Algorithm 3: Number of Connections with Similar Payload-size**

In every data structure in the Spam Relay Database, there are seven parameters in columns for **Number of Connections with Similar Payload-size**. Pick up all the parameters that the value is not equal to 0 in the Spam Relay Database, and the 95th percentile value of these parameters is picked out as the threshold ($\boldsymbol{TH_{payload}}$) for **Algorithms 3**.

- **Generating the Thresholds ($TH_{Vol-Current}$ & $TH_{Vol-total}$) for Algorithm 4&5**

Both of these two algorithms use the volume of SMTP connections in a particular period to distinguish between legitimate hosts and spam relay hosts.

Algorithm 4 uses the volume of SMTP packets with the SYN flag set in the current interval to identify the suspicious spam relay. Assume that the current interval is the N^{th} interval. The connections attempted to establish by each spam relay host are closely correlated to the packets with SYN flag set sent by it. Therefore the number of packets with SYN in the N^{th} interval generated by each host in the Spam Relay Database, which can be obtained from the columns for the **Number of Packets with SYN in N^{th} Interval**, corresponds to the number of SMTP connections that the host attempted to establish. We pick up the number representing the 95th percentile of numbers of SMTP packets with the SYN flag set that was sent out by the hosts in Spam Relay Database in this time interval as the threshold ($TH_{Vol-Current}$) for **Algorithm 4**.

The volume of SMTP packets with the SYN flag set in the most recent 24-hour interval is used to detect spam relays by **Algorithm 5**. The total number of packets with the SYN set for every host in the Spam Relay Database can be calculated by using the following formula.

$$\textbf{Sum} = \sum_{i=1}^{24} \textbf{Parameter in Number of Packets with SYN in i}^{\textbf{th}} \textbf{ Interval}$$

The threshold ($TH_{Vol-total}$) for **Algorithm 5** is set as the 95th percentile number of SMTP packets with the SYN flag set that spam relay hosts send out in the most recent 24-hour period in the Spam Relay Database.

- **Generating the Threshold (TH_{out-in}) for Algorithm 6: Ratio of Out/In SMTP Connection**

Each data structure has a parameter in the column for **Ratio of Out/In SMTP Connection**, which indicates the situation about the response rate of spam emails sent by the host. The 95th percentile of the ratios of Out/In SMTP Connections in Spam Relay Database is used as this threshold($\boldsymbol{TH_{out-in}}$).

- **Generating $\mathbf{D_{threshold}}$ for the Final Decision in the Classifier.**

There is a parameter named **D** in a column in the data structure in the Spam Relay Database, and **D** is generated by the Classifier as a numeral value of the decision. $\mathbf{D_{threshold}}$ is 95th percentile of these values in the Spam Relay Database.

5.2.5.3 Automatic Data Update in System

Updates help the system to automatically do its operations and improve performance. The automatically updating work in our system can be divided into two categories: the first category is the real time updating work, and the second category is the fixed time updating work.

- **Real Time Updating Work**

Most real time updating work is about the generations of the SMTP Traffic Database and Spam Relay Database. When a SMTP packet is logged by the Sniffer, the TCP/IP header information will be used to update the SMTP Traffic Database. The corresponding parameters will be changed in the SMTP Traffic Database. When a host is identified as a spam relay by the Classifier, the information about the host will be used to update the Spam relay Database. Real time updating helps the system to generate and manage both of these two databases.

The coordinates (FIN/SYN flag set, SYN) are also updated in real time. When a host is identified as a spam relay, the coordinate corresponding to this host will be added.

This real time updating helps the system to keep track of the situation in the network.

Also it is able to help system to catch the most recent SMTP characteristics of the legitimate sites and spam relay hosts in the network. It should improve the performance of system.

- **Fixed Time Updating Work**

Fixed time updating work includes two parts: fixed time updating on the database, and fixed time updating of the parameters and thresholds of the system.

One of the purposes of the fixed time updating on the databases is to remove the time expired data structures in each database. It reduces the storage requirements of the system. It could also help system to understanding the most recent situation about the network and spam activities in the network. It improves the efficiency of the system. As it has been said in Chapter 2, the most important periods of time are the day (24 hours) and the week (7 days). SMTP traffic from a legitimate email server, whose SMTP traffic is related well to the time of day, is also related to the day of the week. In order to reduce the requirements of storage size and improve the operational efficiency of the system, system will removed the expired data structure, in which the corresponding IP address has kept silent for over 7 days. These expired data structures are identified by checking the parameters in the column for **Time of Last Packet Arriving** in the structure.

The other purpose of fixed time updating work is about the generation of system parameters and thresholds, such as thresholds for triggers, thresholds for **Algorithm 2**, **Algorithm 3**, **Algorithm 4**, **Algorithm 5** , **Algorithm 6**, and $\mathbf{D_{threshold}}$. A new series of parameters and thresholds help the system to operate automatically more stably and accurately. Because they present the most recent related information of network, they improve the performance. For **Algorithm 1,** a new group of coordinates, in which the expired coordinates have been removed, will be generated by using a fixed time updating process every day. The expired coordinates correspond to the expired hosts, which have been removed from the Spam Relay Database.

The fixed updating time could be set to the quietest period in a day by the administrators. This would improve the efficiency, because the other work is using less computing sources. In our system, the updating time is set to 0:00 every day.

5.2.5.4 Manual Data Update in System

The previous sections introduced the automatic data update scheme in the system. A manual data update scheme is also designed for administrators to check the decision results and adjust the performance of this system,

Administrators of the system can log in the SMTP Traffic Database to know the recent situation of the monitoring network. Also they can log in the Spam Relay Database to check the decision results of the system. If an administrator does not agree with an identification decision to a suspicious host, he can change the decision. When a host is identified as a spam relay by an administrator, this host will be put into Spam Relay Database. Otherwise the host will be removed from the Spam Relay Database.

Performance of the system can be evaluated by administrators via checking SMTP Traffic Database and Spam Relay Database. Some parameters and thresholds (e.g. percentile value for threshold) can be adjusted by administrators to make system meet the security requirements of the network.

Manual data update scheme is designed for administrators to evaluate and adjust the performance of the system. It ensures that the system can avoid some “machine mistakes” to provide a satisfied performance via the effective management of administrators.

5.3 Autonomous Detection System Structure

Figure 5.4 shows the diagram of our autonomous spam relay detection system. The following sections will explain the system's detection processing step by step by reference to this flow chart.

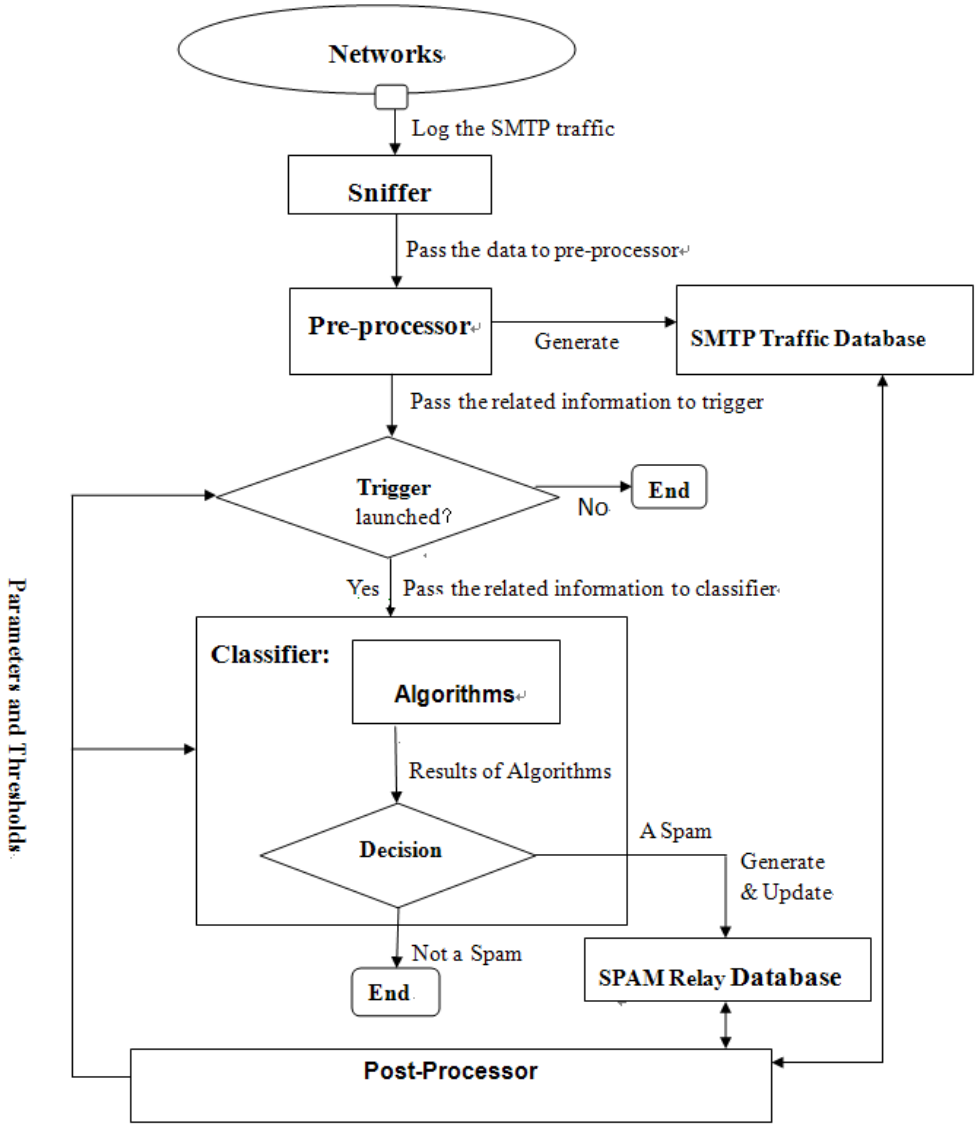

Figure5.4: Diagram of Autonomous Spam Relay Detections System

When the system is installed in a network by administrators, it will start to work automatically.

Firstly, the Sniffer logs the relevant information from the SMTP traffic in the network. The Sniffer captures the SMTP packets and records the TCP/IP Header information of each packet captured. The original information logged by the Sniffer includes source IP address, destination IP address, payload size, flag sets, and so on.

The Pre-Processor processes the original information handed in by the Sniffer and generates the SMTP traffic data structure, in which the data is ready for the measurements of the identification. Each active host in the network is given a corresponding data structure. All of these data structures build up the SMTP Traffic Database. If the source IP of the current SMTP packet arrived is already in the SMTP Traffic Database, the header information of this packet will be used to update to the corresponding columns in the related data structure. Otherwise a new data structure corresponding to this IP address will be created to keep the information.

Thirdly, the Trigger keeps monitoring the updated SMTP Traffic Database. When a new data structure is created or an existent one is updated, the trigger will evaluate the volume of SMTP packets with the SYN flag set in current interval and the volume in the last 24-hour period. If one of these two volumes is above the thresholds of trigger system, the trigger will identify the host, which corresponds to the data structure in SMTP Traffic Database, as a suspicious host. Then the related information of this host in the SMTP Traffic Database will be handed to the Classifier. The Trigger launches the identification process of the classifier.

Fourthly, the classifier identifies whether the host is a spam relay by using the related information passed from the SMTP Traffic Database. Each algorithm in the Classifier will make a decision independently and generate a numerical result. Then, the classifier will make a final decision from these results.

Fifthly, if the host is not thought to be a spam relay by the Classifier, the identification processing finishes and the system returns to standby for the next arrival. Otherwise,

when a host is identified as a spam relay host, the information corresponding to the host in the SMTP Traffic Database will be passed to the Post-processor. The Post-processor will update the Spam Relay Database by using the information. Some parameters and thresholds will be updated by the new information as well, such as coordinates (FIN/SYN flag set, SYN).

Finally, when the system finds that it is time to perform fixed updating, it will automatically start the daily updating. It removes the expired data in SMTP Traffic Database and Spam Relay Database, and generates a series of new system operational parameters and thresholds for another 24 hours.

5.4 Summary

An autonomous system for detecting spam relays using the SMTP traffic characteristics was proposed in this chapter. This system includes five elements: Sniffer, Pre-processor, Trigger, Classifier and Post-processor. The Functions of each element have been represented in detail in this chapter. The information that is used to identify spam relays in this system never involves the real email content. The Sniffer only logs the TCP/IP header information from SMTP traffic packets.

Machine learning techniques have been employed, so that the system can do operations and adjust performance of spam relays identification automatically. There are two databases (SMTP Traffic Database and Spam Relay Database) designed to store and manage the related information of active hosts in the network. SMTP Traffic Database keeps the information of every active host, which has generated SMTP traffic in the last seven days. And Spam Relay Database has the information of these hosts which have been identified as spam relays. Both of these two databases can be updated by updating process of the system. And a series of parameters and thresholds that help system to work automatically and improve the identification performance can also be generated automatically using the information in these two databases by the updating process in the system.

The following chapter will determine that the performance of spam relays detection of the system by using results obtained from a series of tests.

Chapter 6: Testing and Results

A proposed autonomous spam relay detection system has been introduced in Chapter 5, where the system detects the spam relay hosts in the network by the characteristics of SMTP traffic. Machine learning techniques have been applied in the system. In this chapter, the system will be trained by a training data set. The trained system will then be tested to indicate the performance of spam relay detection. A series of tests will be conducted and the results will be presented in this chapter.

6.1 Training process of the Autonomous System

System training processes are used in machine learning and artificial intelligence techniques to optimum system performance. Most of the time, such a system can continue run in real time after the initial training, allowing the system to adapt to the new situation and to changes itself. The initial training is able to help the system generate a series of initial parameters, which are necessary to support the automatically operations of the system [125]. Training data sets were used in the training process. A training set is a set of data used in various areas of information science to discover potentially predictive relationships [126]. Training data sets are popularly used in artificial intelligence, machine learning, genetic programming and intelligent systems. In all these fields, a training data set has much the same role and is often used in conjunction with a test data set.

6.1.1 Training Data Set in Training Process

The data in the training data set was part of the SMTP traffic data from a national ISP's local network and the Loughborough University network. Hosts in these networks include legitimate email use clients, legitimate email servers and spam relay hosts.

There are a total of 220 hosts involved in the training data set. Data in the training data set including the SMTP traffic data collected from 200 legitimate hosts in a 24-hour period and the SMTP traffic data collected from 20 spam relay hosts in a 24-hour period. The data from the legitimate hosts was in the same format as the data structure in the SMTP Traffic Database; while the data from the illegitimate hosts was in the same format as the structure in the Spam Relay Database. All SMTP traffic data in this training data set is not involved the SMTP traffic characteristics analysis in the previous chapters.

	Total Hosts	**Legitimate Hosts**	**Spam Relay Hosts**
Training Data Set	220	200	20

Table 6.1: Hosts in Training Data Set

The training data set will be used by the training process to adjust the parameters of the autonomous system. The following subsection will introduce the training process in more detail.

6.1.2 Training Process

The training Process generates the initial **SMTP Traffic Database** and **Spam Relay Database** by using the training data set. Then a series of parameters and thresholds for the operations of system are generated by using these two databases. After the generation of the two databases, system operation parameters and thresholds, the autonomous system will be ready to be used in a real network for detecting spam relays.

- **Generating the Initial SMTP Traffic Database and related Parameters**

All of the data in the training data set was input to the system to generate the initial SMTP Traffic Database. The initial SMTP Traffic Database would then consist of 220 data structures, which have been described in **Figure 5.2** in Chapter 5. Each data

structure corresponds to a host in the training data set. The average numbers of SMTP Packets with the SYN Flag set in each time interval and in the most recent 24-hour period, which had been established by the corresponding host, can be calculated by using the parameters in columns for **Number of Packets with SYN in N^{th}Interval**. These two average numbers will be used as the initial thresholds for the trigger system, as described in Chapter 5. **Appendix 1** shows these thresholds ($T_{current-th}[i]$ and $T_{total-th}$) for the trigger system after the training process.

- **Generating the Initial Spam Relay Database and Related Parameters**

The SMTP traffic data from spam relay hosts in the training data set was input to the system to generate the initial **Spam Relay Database**. After the training process, there are a total of 20 data structures in this database. Each data structure corresponds to a spam relay host in the training data set, and the information of the host will be recorded in the data structure.

A series of parameters and thresholds are generated: the group of coordinates (FIN/SYN Flag Set, SYN), thresholds for **Algorithms 2, 3, 4, 5 and 6**, and $\mathbf{D_{threshold}}$ for the final decision of the Classifier. The algorithms for these parameter generations have been presented in the Chapter 5. The percentile value for the thresholds is set as 95% in the system. Thresholds for algorithms in the classifier, weight values for each algorithm and $\mathbf{D_{threshold}}$, which had been generated after the training process, had been listed in **Appendix 2**.

After the generation of the two databases and the series of parameters by using the training data set, the training process is completed. The two databases and the parameters will then be used to detect spam relays, when the system is applied to a live network. The information from the training data set in the system will be replaced by new information from the real time SMTP traffic in the network as it expires. The following sections in this chapter will present a series of tests by using testing data sets

to show the performance of the trained system.

6.2 Test Data Sets and System for Tests

In the previous section, the autonomous spam relay detection system was trained by a training data set. In this section, a group of individual test data sets will be used to assess the performance of the system. Results obtained from these tests will show the system's ability to detect spam relays, the contributions of each algorithm in the Classifier, and the effect of the percentile value for thresholds on the performance. The percentile value for threshold defines the reference range for thresholds used in the proposed system.

6.2.1 Test Data Sets

The test data sets [127] were from a national ISP's local network and the Loughborogh University network. But these testing data sets were unique to each other and also to the training data set, which was used previously to train the system. In other words, the hosts in each data set are totally different. Hosts in each test data set include legitimate email sites and spam relays.

A total of 4 group test data sets were prepared. The hosts in these test data sets represent approximately 600 legitimate email users, including servers and clients, and 100 spam relay hosts. There were about 700 hosts in total, and were randomly divided into 4 individual groups. The Data in the test data set was in the same format as the data structure in the SMTP Traffic Database. Each data structure was selected and classified into spam and non-spam categories by manual inspection. **Table 6.2** shows the number of legitimate hosts and spam relay hosts in each testing data set after the manual inspection. Figures in this table will be compared with the results of the tests to show the performance of our spam relay detection system.

Test Data Sets	Total Hosts	Spam Hosts	Legitimate Hosts
Test Data set 1	50	5	45
Test Data set 2	100	20	80
Test Data set 3	200	20	180
Test Data set 4	200	50	150

Table 6.2: Hosts in Test Data Sets

6.2.2 Autonomous System for Tests

The autonomous system for tests in this chapter has already been trained by the training data set in the previous Section 6.1. So the system has already two initial databases (SMTP Traffic Database and Spam Relay Database) and a series of initial parameters and thresholds, which have been shown in Appendix 1 and Appendix 2. The percentile value for threshold is set as 95% in this system.

Because the data in the test sets has already been formatted into the structure used by the SMTP Traffic Database, the Pre-processor will be disabled. The system will then log the data structure into the system, and the trigger process will compare the number of SMTP packets in each time interval and the sum of these numbers with the corresponding thresholds ($T_{\text{current}-\text{th}}[i]$ and $T_{total-th}$). If one of the numbers is above the corresponding threshold, this data structure will be passed to classifier for

identification. The Classifier will identify whether the host is a spam relay by using the identification process described in Chapter 5, when a host's data structure is handed over by the trigger. If a host is identified as a spam relay, the corresponding information will be used to update the Spam Relay Database. SMTP Traffic Database in the testing system is updated by the logging of each data structure in the test data sets.

6.3 Testing Processes and Results

Four groups of tests were designed to achieve the aims of the testing process. The first group of tests was used to evaluate spam relay detection performance. The second group of tests was to assess the contribution of each algorithm in the classifier. And the third group was for identifying how the percentile value for thresholds affects the performance of system. Test results of these three groups of tests have been listed in Appendix 3. Results in Appendix 3 included outputs of each algorithm and the value of D (final decision) of each host which has been identified as a spam relay by our detection system .The last group of tests were used to indicate the performance of the update process in the detection system.

6.3.1 Evaluating the Ability to Detect Spam Relays

Each test data set was logged by the system, in which the percentile value was set as 95%. Before a new data set was logged, the system was reset to the initial status, which is the status after the training process

Test results, which including weight of values, $\mathbf{D_{threshold}}$, outputs of each algorithm for each spam relays detected by the system, and value of final decision D for each spam relays detected by the system, were listed in Appendix 3.

Table 6.3 shows the results of spam relay detection by using these test date sets

Test Data Sets	Total Host	Spam Relays	Spam Detected	False Positives	False Negatives
Test Data Set 1	50	5	5	0	0
Test Data Set 2	100	20	17	0	3
Test Data Set 3	200	20	19	1	1
Test Data Set 4	200	50	41	0	9

Table 6.3: Results of the Tests by Using Test Date Sets

The maximum rate of spam relay detection was 100% in **Table 6.3**, when the system was tested by using Test Data Set 1. The minimum rate is 82%. On average, 91% of the spam relays were positively identified by the autonomous system. There was only one false positive error in the four tests. The average ratio of false positives error is around 0.13%.

It was also found that the spam relay detection rate increased as the percentage of spam relays hosts in the network decreased. The percentage of the spam relays is lower than 10% in **Test Data Sets 1 & 3**, and the ratio of spam relay detection corresponding to these two test data sets are 100% and 95%. But in the **Test Data Set 4**, the identification rate is only 82%, and the percentage of spam hosts in this data set is 20%. More percentage of the spam relays may be identified in a network with fewer spam relays. There is a false positive error occurred in spam relay detection process by using the **Test Data Set 3**. This unique false positive error occurred in the testing, in which system was tested by using the test date set with the lowest percentage of spam relay hosts. It may suggest that false positive errors may be introduced into the spam relay identification, when this system is applied in a network with fewer spam hosts.

Every anti-spam technique has trade-offs between the false negative and false positive responses. These days, a spam filter is considered effective if it has a detection rate 90%

or high [128]. So we are able to say that the proposed system has a good performance for detecting the spam relays in networks. It has a high positive detection rate with a low false positive rate.

6.3.2 Assessing the Performance of Each Algorithm in the Classifier

In the proposed autonomous spam relay detection system, the Classifier is composed of six algorithms whose weighted outputs are combined together to produce the overall result (D) seen in Chapter 5. Each algorithm was designed according to the results of the analysis of the SMTP Traffic characteristics for legitimate and illegitimate sites in Chapter 4. In this section, the results from test processes will indicate the contribution of each algorithm to spam relay identification of the Classifier. Some other results from these test processes will show the necessity of combining a variety of algorithms in the Classifier to achieve a satisfying performance.

The system recorded outputs of each algorithm and final decision values (D) for every spam relay hosts that were identified by our system in the test processes. Outputs of each algorithm and final decision values have been listed in Appendix 3. In Chapter 5, it was stated that each algorithm would generate a result with the value 1 if a host is thought to be a suspicious spam relay. According to this, it is easy to find which host was identified as a suspicious spam relay by an individual algorithm.

Table 6.4 shows the spam relay identification result for each individual algorithm in the Classifier in the test process, in which **Test Data Set 3** was used. **Test Data Set 3**, which includes 180 legitimate hosts and 20 spam relay hosts, was used in this test. The value of percentile for thresholds was 95% in this test system, as before.

Algorithms	Name of Algorithms	Value of weight for Algorithm	Spam Hosts in Test Data	Spam Hosts Detected	False Positives	False Negatives
Algorithm 1	**Ratio of FIN/SYN Flag Set**	0.888889	20	7	4	13
Algorithm 2	**Relative to Time of Day**	0.684211	20	17	10	3
Algorithm 3	**Similar Payload-size Connections**	0.666667	20	15	0	5
Algorithm 4	**Volume of Connections in Current**	0.555556	20	20	16	0
Algorithm 5	**Volume of Connections in Recent 24**	0.642857	20	17	6	3
Algorithm 6	**Ratio of OUT/IN SMTP Connections**	0.863636	20	20	2	0

Table 6.4: Identification Results of Individual Algorithm in Classifier in the Test Process by Using Test Data Set 3

Table 6.4 shows that each individual algorithm in the Classifier is able to pick up a number of spam relays. However the identification rate and false positive and negative rate are not as good as expected. The average identification rate is about 80%, and the minimum rate of these algorithms is only 35%. **Algorithms 4 and 6** have the high identification rates, however the false positive rates of both these algorithms are larger than a system in which 6 algorithms combined. **Algorithm 4** has 16 false positive errors in decisions of 200 hosts. The average false positive error rate is 3.17%. The greater the percentage of false positive errors is in the decisions, the greater the harm done to legitimate users in the network. In summary, any one algorithm in the Classifier can pick out a number of spam relays, but is not enough to identify spam relay hosts with satisfactory performance.

A total of six algorithms are combined in the classifier. The performance of each individual algorithm for the spam relay detection has been discussed in the previous section. The following sections will discuss the performance of the system, in which only some of these algorithms are enabled. The results from the disabled algorithms were set as a constant 0. And then a new $\mathbf{D_{threshold}}$, which only considers the results from the enabled algorithms, had to be generated by training process before the system was tested. Test Data Set 3 was used for the test processes of these systems in which only several algorithms were enabled. Test results, which included outputs of each enabled algorithm, weights values of algorithms, value of $\mathbf{D_{threshold}}$ and values of final decisions (D), have been listed in **Appendix 4**.

Table 6.5 shows the spam relay identification results of the test processes by using the **Test Data Set 3** on these systems, in which only several algorithms were enabled and the percentile value was set as 95%.

Enabled Algorithms	Spam hosts in Test Data Set	Spam Hosts detected	False Positive	False Negative
Algorithm 1, 3&5	20	15	3	5
Algorithm 1, 3&6	20	15	3	5
Algorithm 3, 4&5	20	15	0	5

Table 6.5: Test Results from the System with Several Algorithms Enabled

Table 6.5 shows that a system, which uses only several algorithms in its classifier, can also pick up some of the spam relay hosts. The spam relay identification rates of these three systems are the same (75%). It shows that they can provide a more stable identification performance than an individual algorithm. Also the situation of the false positive error rate is improved, when more algorithms are used in the system. The average false positive rate is 1.00%, and the maximum rate is 1.50%. It was also found that the selection of algorithms could affect the performance of the system.

In summary, it was found that:

1. A classifier using just one detection algorithm would not be good enough to be used for spam relay host detection. It may be possible to pick up a number of spam relays, but it would also produce many false positive errors, which would do harm to the legitimate users.

2. That additional algorithms used in classifier reduce the false positive error rate.

3. The selection of algorithms, which are enabled in the classifier, affects the performance of spam relay detection.

6.3.3 Effect of the Percentile Value for Thresholds on the Performance of the System

In our system, the percentile value for the thresholds is used to define the reference range of the related measurement. The thresholds for **Algorithms 2, 3, 4, 5&6** and final decision of the classifier are generated by using this coefficient. It is found that all these thresholds are the lower bounds of the reference ranges. So when the percentile value is reduced, these lower limits are increased, which will make the identification rules more strict. This coefficient (percentile value) can also be set by the network administrators. It is probably changes of the percentile value affect the performance of the system. A group of tests was designed to investigate this issue.

Table 6.3 has shown the test results from the system with a percentile value set as 95%. Then the system, in which this percentile value was set to 50%, was trained by the same training data set. After the training process, the system was tested by the same four test data sets. Test results, which included outputs of each algorithm, weight values and final decision value D, have been listed in Appendix 3. **Table 6.6** shows the test results from the system with the percentile value set to 50%.

Test Data Sets	Total Host	Spam Relays	Spam Detected	False Positives	False Negatives
Test Data Set 1	50	5	3	0	2
Test Data Set 2	100	20	10	0	10
Test Data Set 3	200	20	9	0	11
Test Data Set 4	200	50	17	0	33

Table 6.6: Test Results of Spam Relay Detection by Using the System with Percentile Value 50%

It is found that the spam relay identification rate of every test data set was reduced in comparison with the corresponding rate in **Table 6.3**. The average positive spam relay identification rate is only 47.25%. In other words, there are only 47.25% of the spam relays in the test data sets could be identified. However not everything is disappointing. The results in **Table 6.6** show that there are no false positive errors generated by this system in any of the tests.

We are therefore able to say that the percentile value for thresholds affects the performance of the system. A reduction of this value can reduce the positive spam relay identification rate, but also reduce false positive errors.

6.3.4 Performance of the Update Process in the Proposed System

In this section, the performance of update process will be presented. A group of tests were conducted to evaluate the performance of the update process. The proposed system, in which the percentile value was set as 95%, was trained by the training data set firstly. Then the system, which had been already trained, was used to test by using **Test Data Set 3**. After this test, new databases (SMTP Traffic Database and Spam Relay Database) have been generated. Then the update process was launched. Update process in this system generated new thresholds and parameters for the proposed system by using the information from new databases. These new thresholds and parameters for the system had been listed in Appendix 5.

Test data Set 1, Test Data Set 2 and Test Data Set 4 were used to test the performance of the system, which has the new thresholds and operation parameters after the update process. Test results have been listed in Appendix 6, in which there are outputs of each algorithm, final decision value, weight values and so on. **Table 6.7** shows the spam relay identification results of this group of test processes by using the system after the update process.

Test Data Sets	Total Host	Spam Relays	Spam Detected	False Positives	False Negatives
Test Data Set 1	50	5	5	0	0
Test Data Set 2	100	20	17	0	3
Test Data Set 4	200	50	42	1	8

Table 6.7: Results of the Tests on the Proposed System after Update Process by Using Test Date Sets

The maximum rate of spam relay detection was 100% in **Table 6.7**, when the system was tested by using Test Data Set 1. The minimum rate was 84%. 90% of the spam relays were positively identified by the autonomous system on average. There was only one false positive in the four tests. The average ratio of false positives was around 0.17%. Results from Table 6.3 and Table 6.7 tell that both the system before the updating and the system after the updating have high spam relay identification rates (over 90%) and low false positive error rates (less than 0.2%).

So we are able to say that the update process, which was designed for the detection system, is suitable for the proposed system, because the system can still provide a high positive detection rate with a low false positive rate after the update process.

6.4 Conclusions of System Tests

In this chapter, the autonomous spam relay detection system was trained by a training data set. The training data set helped the system to generate the initial SMTP Traffic Database, the initial Spam Relay Database and a series of initial operational parameters and thresholds.

The system was tested with 4 groups of test data after the training process. A series of tests were done to evaluate the performance and understand the influence of the changes to the system. A lot of results and analysis have been mentioned in this chapter, and are summarized below:

1. The autonomous system is able to identify spam relays, after it has been trained. The positive spam relay identification rate of the system, in which the percentile value for thresholds was set as 95%, reached on average number of 91%. At the same time the ratio of false positive errors was about 0.13%.

2. There are six algorithms combined in the Classifier. These algorithms are designed according to the analysis of the SMTP traffic characteristics performed in Chapter 4. Test results show that every algorithm gives a contribution to the spam relay identification of the classifier. On one hand each individual algorithm can pick up some of spam relays, on the other hand each individual algorithm could make more mistakes in the identification, which would do much harm to the legitimate users. An individual algorithm is not enough to provide good spam relay identification.

3. Test results also indicated that combinations of the algorithms in the classifier could improve the performance. The obvious improvement is on the reduction of the false positive rate. Different selections of the algorithms also affected the performance of the system. However the number of algorithms combined in the classifier would be limited in practice due to complexity of the system

4. The percentile value for thresholds in the system can affect the performance of the system. A reduction of this value can reduce the positive spam relay identification rate, but it can also reduce the false positive error rate. This percentile value can be set by network administrators to meet the security requirements of the network under protection.

5. The update system in the proposed system is suitable for keeping the performance of spam relay identification as well as expected. A series of tests have been conducted to present performance of the system in which the update process have been launched before the proposed system do next job of the spam relays detections. The positive spam relay identification rate of this system still reached on average number of 90% in that situation. At the same time the ratio of false positive errors was 0.17% on average.

Chapter 7: Conclusions and Further Work

Spam emails are flooding the internet and do great harm to people every day. A lot of anti-spam techniques have been developed to fight with spam emails. But so far the volume of spam emails is still increasing. In this thesis, the definition of spam emails, the harm done by spam emails, spammer activity and anti-spam techniques were reviewed. SMTP traffic data was collected from real networks. The SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays were analyzed. An autonomous system for detecting spam relays using SMTP traffic characteristics was designed and tested in this research. The following sections will present the results and conclusions of this research.

7.1 Results and Conclusions

Spam emails are unsolicited bulk emails, and spammers send mass spam emails to achieve their selfish purposes. Spam email cost people's money and time, degrades the performance of networks, and also causes a large amount of security problems for networks.

Spammers harvest as many email addresses as they can. They compose spam emails that are more likely to capture the recipients' attention and resort to re-routing spam emails through relay hosts in networks to avoid detection. Spam tools are developed and applied for automatic spam activities by spammers.

A large number of various anti-spam techniques have been developed and employed to prevent spam emails. However all this work is not enough, the problem caused by spam emails continues to become more and more serious. The general consensus is that a single technical solution that is able to prevent the propagation of spam is

unlikely to be found given the constraints of the current Internet architecture. In this thesis, some commonly used anti-spam techniques have been discussed, e.g. DNSLs, Bayesian Spam Filter, and Checksum Based Filters and so on.

SMTP traffic has been collected from the Loughborough University campus network and a national ISP's network which is one of the UK national wide ISP local networks. SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays were analyzed and compared. It has been shown that legitimate email clients, legitimate email servers and spam relays have their own characteristic SMTP traffic profiles. These differences regarding the SMTP traffic characteristics result from characteristics such as volume of SMTP connections (number of emails sent out), successful connection rate (not-reject rate), payload size of each connection, response rate (the rate of reply to email), and the relationship between the traffic and time of day, and so on. A legitimate email client is seen to typically establish several SMTP connections over hours to one or a few email servers. A lot of connections, in which the sizes of payloads are different most of time, are established by a legitimate email server. SMTP traffic from a legitimate email server follows a regular time of day profile. By Contrast, spam relay hosts send a mass of similar emails to a huge number of destination addresses in a session for a lot of time. The SMTP traffic from the spam relays is also not well related to time of day most of time in most cases. Cyclical and periodic phenomena often appear on the profiles of the spam relay's traffic. Also some spam relays attempts to establish SMTP connections are rejected, and there is little incoming traffic to a spam relay host.

In this thesis, the understanding of the SMTP traffic characteristics from different sources (legitimate email clients, legitimate email servers and spam relays) suggested that some methods and parameters might be used to identify spam relays in a network. Spam relays may be able to identified by evaluating the successful connection rate via the ratio of the FIN/SYN flag set, by counting the total number of the connections in a particular time interval, by comparing the size of the payload in each connection, by

evaluating the ratio of OUT/IN SMTP packets with the SYN flag set, or by evaluating the relationship between the SMTP traffic and time of day.

An autonomous system for detecting spam relays by using these SMTP traffic characteristics was proposed in this thesis. This system included five elements (Sniffer, Pre-Processor, Trigger, Classifier, and Post-Processor) and two databases (SMTP Traffic Database and Spam Relay Database). Methods and parameters which have been mentioned in the previous section were combined into the Classifier in the system. It is important to note that the information that is used to identify spam relays in the network never involves email real content. The Sniffer only logs the TCP/IP header information from the SMTP traffic packets. Machine learning technologies have been employed in the system, so that the system is able to operate automatically and improve the identification performance via an updating process.

A series of tests have been conducted to determine the performance of the system. The following subsections represent the results obtained from these tests.

1. After training, the system is able to identify spam relays in the network. The spam relay identification rate could reach 91% on average, and the rate of false positive errors is 0.13% on average.

2. Each algorithm, which is combined into the classifier, provides a contribution to the spam relay detection. An individual algorithm is not enough to successfully identify spam relays. The Combination of algorithms improved the identification performance of the system.

3. The percentile value for thresholds has been seen to affect the performance of the system. The reduction of this coefficient can reduce the rate of spam relay identification rate, but it also reduces the false positive error rate.

4. False positive errors vs. false negative errors are still a problem for the spam relay detecting system. However choosing an appropriate threshold percentile could help the system to meet the requirements of the monitored network.

5. The update process designed for the proposed system is suitable for keeping the performance of spam relay identification as well as expected.

7.2 Summary of Contributions

A way to collect a set of data from live networks was found. A sniffer was created in the C programming language to collect SMTP traffic data from a real network. SMTP traffic data sets were generated. In this research, SMTP traffic data was collected from different sources (legitimate email clients, legitimate emails servers and spam relays) in real networks (A national ISP's network and Loughborough University campus network).

The differences regarding SMTP traffic characteristics of legitimate email clients, legitimate email servers and spam relays were determined by analyzing SMTP traffic that collected from live networks. It was found SMTP traffic from legitimate sites and illegitimate sites were different and could be distinguished from each other by using SMTP traffic characteristics. Some methods and parameters based on analyzing SMTP traffic characteristics were proven to be able to identify spam relays in the network.

Another contribution of this research is in developing an autonomous system for detecting spam relays by using SMTP traffic characteristics. This proposed system identifies spam relays in real time before spam emails get to an end user. It is important to note that the information that is used to identify spam relays never involves email real content. Machine learning technology was employed in this system. Results obtained from the tests show that this system has a high spam relay detection rate and an acceptable false positive error rate.

A contribution that must not be neglected is the finding that a combination system is able to provide better performance of spam relay detection. An individual algorithm in the Classifier can pick up a number of spam relays, but it is no good enough to be used for spam relay detection. A series of tests were conducted to state that a classifier that employed additional algorithms has more opportunities to provide good performance than a classifier using just one detection algorithm. The most significant result of our

research is that selection of algorithms focused on different independent area of SMTP characteristics provides a high spam relay detection rate and an acceptable false positive error rate.

The last contribution of this research work is finding a way to adjust the performance of the system to meet the security requirements of the network under protection. It was found that the change of percentile value for thresholds in the system could affect the performance of the system. This percentile value can be set by network administrators to meet the security requirements.

7.3 Further Work

As was stated in Chapter 1, a single technical solution that is able to prevent the propagation of spam is unlikely to be found in the current Internet architecture. So currently, combination of systems is popularly employed to fight spam emails. In this thesis, it was shown that different combinations of algorithms in the classifier provided different levels of performance of identification. Further research could include two issues: one is finding more methods that can be combined in this system to identify spam relays, and the other is finding a way to build up a combination system with a better performance, which involves suitable choices of other anti-spam methods that could be combined in the system.

Currently, there are many anti-spam techniques employed to fight spam emails in the internet. Some techniques could be combined into the Classifier in this system as an individual algorithm. Therefore, a combination system, in which different types of techniques are combined, could be developed. A series of tests would be conducted to determine the performance of the new combination systems. Results from these tests not only may help to evaluate the performance of the systems, but also evaluate the ability of each individual technique to identify spam relays. This work may help to find more techniques that are able to be combined in this system to achieve better performance of spam relay identification.

When the system was being designed, consideration was given to its complexity and efficiency. Therefore, it is very important to know how to choose the most suitable anti-spam methods to be combined in the system. In the process of building and evaluating combination systems, it would be useful to find some rules to predict the suitable anti-spam methods, which are in a combined system, will improve the performance.

Reference

[1] Royal Pingdom, “Internet 2010 in Numbers”, 12 January 2011. Retrieve 2011-10-25, from http://royal.pingdom.com/2011/01/12/internet-2010-in-numbers/.

[2] Microsoft, “Microsoft Security Report”, 8th April 2009.

[3] Spam Task Force Network and Technology Working Group, “Anti-spam Techniques Overview”, Telecommunications Engineering and Certification Industry Canada, May 2005, Page 9.

[4] Scott Hazen Mueller，“What is spam?”, Spam.abuse.net, Retrieved 2011-8-21, from: http://spam.abuse.net/overview/whatisspam.shtml, Retrieved 2011-8-21.

[5] Infinite Monkeys & Company , “Spam Defined”, Retrieved 2011-8-21, from: http://www.monkeys.com/spam-defined/ .

[6] James John Farmer, “An FAQ for news.admin.net-abuse.email, Part 3: Understanding NANAE”, 27 December 2003. Retrieved 2011-8-21 from: http://web.archive.org/web/20040212175535/http://www.spamfaq.net/terminology.shtml.

[7] Brad Templeton, “Essays on Junk Emails (Spam)”, templeton.com. Retrieved 2011-10-26, from: http://www.templetons.com/brad/spam/

[8] Carl Eklund,”Spam -from nuisance to Internet Infestation”, Peer to Peer and SPAM in the Internet Raimo Kantola’s technical report, 2004, Pages 126-134.

[9] Spam Laws, “The Cost of Spam: Financial Risks”, Retrieved 2011-10-25, from: http://www.spamlaws.com/spam-financial-risks.html.

[10] Darren Waters, “Spam overwhelms e-mail messages’, BBC News Website, 8th April 2009. Available from: http://news.bbc.co.uk/1/hi/technology/7988579.stm, Retrieved 2011-10-25.

[11] Jose Norte Sosa, “Spam Classification Using Machine Learning Techniques – Sinespam”, Master of Science Thesis, University Politècnica de Catalunya, August 2010. Page 10.

[12] S. Hird, “Technical Solutions for Controlling Spam”, in Proceedings of AUUG2002, Melbourne, September 2002. Page 3.

[13] Henry Stern, “A survey of Modern Spam Tools”, 5th Conference on Email and Anti-Spam, CEAS2008, Mountain View, California, Aug 21-22, 2008. Pages 2-4.

[14] Pedro Calais, Dorgival Guedes, Wagner Meira Jr., Cristine Hoepers, Marcelo Chaves and Klaus Steding-Jessen, “Spamming Chains: A New Way of Understanding Spammer Behavior”, 6th Conference on Email and Anti-Spam, CEAS2009, Mountain View, California, July 16-17, 2008.

[15] Anirudh Ramachandran, Nick Feamster, ”Understanding the network level behaviour of spammers”, SIGCOMM 06,Pisa, Italy, 2006. Pages 291-302.

[16] Ahmed Obied, “Honeypots and Spam”, Department of Computer Science University of Calgary. Page 7.

[17] KJ Beer, “SYSTEM AND METHOD FOR PREVENTING THE RECEPTION AND TRANSMISSION OF MALICIOUS”, US Patent App. Application Number: 12/117,847, Publication Number: US 2008/0282338 A1, Filing data: May 9, 2008.

[18] Mark Levitt & Brain E. Burke, “Choosing the Best Technology to Fight Spam”, IDC white paper, April 2004. Pages 5-7.

[19] Joon S. Park, Hsin-Yang Lu and Chia-Jung Tsui, “Anti-spam approaches: analyses and Comparisons”, The Open Information Systems Journal, 2009.

[20] Barracuda Networks, “An Overview of Spam Blocking Techniques”, Report from Barracuda Networks, 10040 Bubb Road, Cupertino, CA 95014. Pages 1-7.

[21] Tan Ying & Zhu Yuan-chun, “Advances in Anti-spam Techniques”, CAAI Transactions on Intelligent Systems, 2010 Issue 3, June 2010. Pages 189-201.

[22] MX Logic, “Spam Classification Techniques”, Report from MX Logic, 9780 Mt. Pyramid Court, Suite 350, Denver, Co, 80112 USA, 2004. Pages 2-7.

[23] Prashanth Srikanthan, “An Overview of Spam Handling Techniques”, Computer Science Department, George Mason University, Fairfax, Virginia 22030, 2003, Posted by Gusaul “Founder”, July 26 2012.

[24] Anselm Lambert, “Analysis of Spam”, Master of Science Thesis, Department of Computer Science, University of Dublin, Trinity College, September 2003.

[25] P J Sandford , J M Sandford, and D J Parish, “Analysis of SMTP Connection Characteristics for Detecting Spam Relays”, International Multi-Conference on Computing in the Global Information Technology ICCGI06, 2006. Pages 2-3.

[26] Tim Weber, “Gates forecasts victory over spam”, BBC Online Business News, 24th January 2004. Retrieved 2011-10-25, from: http://news.bbc.co.uk/1/hi/business/3426367.stm

[27] Techgurulive, “Spam level *declines*… to 97 percent of all email”, www.techgurulive .com. Retrieved 2011-10-25, available from: http://techgurulive.com/2009/04/14/spam-level-declines-to-97-percent-of-all-email/

[28] Maria Namestnikova, August 2010, “Spam Report: August 2010”, the article from Securelist. Retrieved 2011-10-25, from: http://www.securelist.com/en/analysis/204792138/Spam_Report_August_2010.

[29] Computer service Loughborough University, “Increase in junk email volume”, Report from Loughborough University IT Service Group, October 2010.

[30] Nicola Lugaresi, “European Union vs. Spam: A Legal Response”, 1st Conference on Email and Anti-Spam, CEAS2004, Mountain View, California, July 30-31, 2004.

[31] RFC 821, “Simple Mail Transfer Protocol”, J.B. Postel, the Internet Society, August 1982.

[32] RFC 5321, “Simple Mail Transfer Protocol”, J. Klensin, the Internet Society, October 2008.

[33] Tamara Dean, “Network + Guide to Networks” (fifth edition), Course Technology, ISBN-10: 1423902459, March 2009. Page 519

[34] RFC 3501, “Internet Message Access Protocol 4rev1”, M. Crispin, the Internet Society, May 2003.

[35] Spam Task Force Network and Technology Working Group, “Anti-spam Techniques Overview”, Telecommunications Engineering and Certification Industry Canada, May 2005, Page 8.

[36] RFC 5782, “DNS Blacklists and Whitelists”, J.Levine, Internet Research Task Force, Taughannock Networks, February 2010.

[37] S. Hird, “Technical Solutions for Controlling Spam”, September 2002, in Proceedings of AUUG2002, Melbourne. Page 5.

[38] Jaeyeon Jung, Emil Sit, “An Empirical Study of Spam Traffic and the Use of DNS Black lists”, 4th IMC 2004, in Taormina, Sicily, Italy, October 25-27, 2004.

[39] DNSBL.info, “Spam Database Lookup”, dnsbl.info. Retrieved 2011-10-26, Available from: http://www.dnsbl.info/dnsbl-list.php.

[40] Spamlinks, “DNS & RHS Blackhole Lists”, spamlinks.net. Retrieved 2011-10-26, Available from: http://spamlinks.net/filter-dnsbl-lists.htm.

[41] Active Web Hosting. “Spam Keywords to Add to Your Filter Lists”. Retrieved on 2011-11-17, from: http://www.activewebhosting.com/faq/email-filterlist.html

[42] Lin Wei, Department of Computer Science, Sichuan Police College, **“**A Bayesian Spam Filtering Method Based on Words Probability”, “Computer Technology and Development 2011-09”, September 2011.

[43] Aris Kosmopoulos, Georgios Paliouras and Ion Androutsopoulos, “Adaptive Spam Filtering Using Only Naive Bayes Text Classifiers”, 5th Conference on Email and Anti-spam, CEAS 2008, Mountain View, California. Aug 21-22, 2008.

[44] Gumpina V V Satya Prasad, Satya P Kumar Somayajula, “ Bayesian Spam Filtering Using Statistical Data Compression”, in “Global Journal of researches in engineering Numerical Methods” Volume 11 Issue 7 Version1.0. Publisher: Global Journals Inc. (USA), Online ISSN: 2249-4596& Print ISSN: 0975-5861. December 2011.

[45] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C.D. Spyropoulos, and P. Stamatopoulos, “Learning to filter spam e-mail: A comparison of a naive bayesian and a memorybased approach”. “Proceedings of the Workshop on Machine Learning and Textual Information Access”, H. Zaragoza, P. Gallinari, and M. Rajman (Eds.), 4th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-2000), Lyon France, September 13-16, 2000, Pages 1-13.

[46] G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C. D. Spyropoulos, and P. Stamatopoulos, “A memory-based approach to antispam filtering for mailing lists” , Information Retrieval, 6, 2003, Pages 49-74.

[47] C. Cortes and V. Vapnik, “Machine Learning: Support-vector networks”, Spring Netherlands, Vol. 20, No.3, September 1995. Pages 273–297.

[48] N. Cristiatnini and J. Shawe-Taylor, “An introduction to Support Vector Machines and Other Kernel-Based Learning Methods”. Published by The Press Syndicate of the University of Cambridge, ISBN: 0-521-78019-5, March 2000. Page 7.

[49] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition”. Data Mining and Knowledge Discovery, Vol.2, No.2, DOI: 10.1023/A: 1009715923555, 1998, Pages121–167.

[50] Enrico Blanzieri , Anton Bryl, “Instance-based spam filtering using SVM nearest neighbor classifier”, Proceedings of the Twentieth International Florida Artificial Intelligence Research Society Conference, May 7-9, 2007, Key West, Florida, USA. Pages 441-442.

[51] Yuewu Shen, “Using Feature Selection to Speed Up Online SVM Based Spam Filtering”, Asian Language Processing (IALP), 2010 International Conference, Harbin, China. December 28-30 2010, Pages 142-145.

[52] A. Ratnaparkhi, “A simple introduction to maximum entropy models for natural language processing”, Institute for Research in Cognitive Science, IRCS Technical Reports Series, University of Pennsylvania, 1997.

[53] Shaohong Zhong, “An effective spam filtering technique based on active feedback and Maximum entropy”, 2010 7th International Conference on FSKD, Vol.5, Print ISBN: 978-1-4244-5931-5, Yantai, China, Aug 10-12, 2010, Pages 2437-2440

[54] Chih-Hung Wu and Chiung-Hui Tsai, "Robust classification for spam filtering by back-propagation neural networks using behavior-based features", Volume 31, Number 2 (2009), DOI: 10.1007/s10489-008-0116-0, Applied Intelligence, 2009, Pages 107-121.

[55] Chih-Hung Wu, "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks", "Expert Systems with Applications" Volume 36, Issue 3, Part 1, April 2009, Pages 4321–4330

[56] J. Holland, "Adaptation in Natural and Artificial Systems", Publisher: A Bradford Book, ISBN-10: 0262581116, Publication Date: April 29, 1992.

[57] Md. Saiful Islam, Shah Mostafa Khaled, Khalid Farhan, Md. Abdur Rahman and *Joy Rahman, "Modeling Spammer Behavior: Naïve Bayes vs. Artificial Neural Networks", 2009 International Conference on Information and Multimedia Technology, Jeju Island, South Korea, December 18-19, 2009, Conference Publication, Print ISBN: 978-0-7695-3922-5, Pages 52-55.

[58] P.Mohan Kumar. P.Kumaresan. S.Yokesh Babu, "Accuracy Analysis of Neural Networks in Removal of Unsolicited e-mails", International Journal of Computer Applications, Volume 16– No.3– Article 7, IJCA journal, Published by Foundation of Computer Science, ISBN: 978-93-80747-57-9, February 2011.

[59] Pedro G. Espejo, Sebastian Ventura, Francisco Herrera, **"**A survey on the application of genetic programming to classification", Journal IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, Volume 40 Issue 2. Publisher: IEEE Press Piscataway, NJ, USA, ISSN: 1094-6977, March 2010. Pages 121-144.

[60] HAJIRA JABEEN* AND ABDUL RAUF BAIG, “Review of Classification Using Genetic Programming”, International Journal of Engineering Science and Technology Vol.2, Issue 2, IJEST(ISSN: 0975-5462), February 2010. Pages 94-103.

[61] M. Sahami, “ A Bayesian Approach to Filtering Junk Email”, In Proceedings of AAAI-98 workshop on Learning for Text Categorization, Madison, Wisconsin, USA, 1998.

[62] Joon S. Park, Hsin-Yang Lu and Chia-Jung Tsui. “Anti-Spam Approaches: Analyses and Comparisons”. The Open Information Systems Journal, 2009, 3. Pages 36-47.

[63] Adeleh Jafar Gholi Beik, Ali Haroun Abadi, and karim Ansari Asl, “Anti Spam Filtering keyword-based and multi agent method with personal E-mail messages on the basis of interests of user”, Canadian Journal on Artificial Intelligence, Machine Learning and Pattern Recognition Vol. 2, No. 4, May 2011.

[64] Triola, Mario F, “Bayes' Theorem”, Elementary Statistics 11th edition, Publisher: Addison Wesley, ISBN10: 0321500245, January 7 2009.

[65] Richard C. Carrier, Ph.D. “Bayes’ Theorem for Beginners: Formal Logic and Its Relevance to Historical Method — Adjunct Materialsand Tutorial”, the Jesus Project Inaugural Conference “Sources of the Jesus Tradition: An Inquiry”, Amherst New York, December 5-7, 2008.

[66] Frederic P. Miller, Agnes F. Vandome, John McBrewster, " Bayesian Spam Filtering", Alphascript Publishing, ISBN6130213492, ISBN9786130213497, July 27 2010.

[67] Brain Livingston, “Paul Graham provides stunning answer to spam e-mails:Probability theory shows impressive results”, InfoWorld, 20th August 2002. Retrieved 2011-11-3, from: http://www.infoworld.com

[68] Paul Graham, “Better Bayesian Filtering”, MIT Spam Conference 2003, Cambridge, Unite States .17th January 2003.

[69] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, George Paliouras and Constantine D. Spyropoulos, “An Evaluation of Naive Bayesian Anti-Spam Filtering”, Proceedings of the workshop on Machine Learning in the New Information Age, G. Potamias, V. Moustakis and M. van Someren (Eds.), 11th European Conference on Machine Learning, Barcelona, Spain, 2000, Pages 9-17.

[70]GFI Software, “Why Bayesian filtering is the most effective anti-spam technology”, GFI White Paper, 2011, Pages 3-4.

[71] Altus Security, “Web and Email Security Solutions”, Retrieved on 2011-11-7, form: http://doublesix-networks.com/security_solutions/url_content_filtering.php.

[72] David Harley, Andrew Lee, “the Spam-ish Inquisition”, ESET antivirus and security white papers, ESET, LLC, Califonia, USA, 2007. Pages 3-4.

[73] Flavio D. Garcia, Jaap-Henk Hoepman and Jeroen van Nieuwenhuizen, “Spam Filter Analysis”, Security and Protection in Information Processing Systems, IFIP International Federation for Information Processing, 2004, Volume 147/2004, Pages 395-410, DOI: 10.1007/1-4020-8143-X_26.

[74] Rhyolite Software, “Distributed Checksum Clearinghouse”, Rhyolite Software LLC. Available from: http://www.rhyolite.com/dcc/, Retrieved 2011-11-7.

[75] Vipul Ved Prakash. “Vipul’s Razor”, 2007. Retrieved on 2011-11-7, Available from: http://razor.sourceforge.net/.

[76] SpamAssassin, “The Apache SpamAssassin Project”, Spamassassin official website. Retrieved 2011-11-8, available from: http://spamassassin.apache.org/

[77] Cloudmark, “Cloudmark Authority-Server Based Spam Remediation Software”. Cloudmark Inc, 500 Third Street, Suite 265, San Francisco, CA 94107-1805. 30th May 2003.

[78] David J. Bilinsky, “Mastering your Mailbox: E-mail and Information Management”, 2nd Annual Solo and Small Firm Cofference and Expo, Toronto, March 2007, Page 14.

[79] Sourceforge.net, “Crm114- the Controllable Regex Mutilator”, Retrieved 2011-11-8, form: http://crm114.sourceforge.net/

[80] Rebecca Lieb, “Make Spammer Pay Before you Do”, The ClickZ Network, Jul 26, 2002. Retrieved 2011-11-8, available from: http://web.archive.org/web/20070807113021/http://www.clickz.com/showPage.html?page=1432751

[81] Michael Ilger J¨urgen Strauß Wilfried Gansterer Christian Proschinger , “The Economy of Spam”, Technical Report FA384018-6, Institute of Distributed and Multimedia Systems, University of Vienna, 12th September, 2006, Page 3.

[82] Ben Laurie and Richard Clayton, “‘Proof-of-Work’ Proves Not to Work,” ALD LTD & University Cambridge, Computer Laboratory. May 3, 2004, Pages 2-8.

[83] Debin Liu, L Jean Camp, “Proof of Work can Work”, The 5th Workshop on the Economics of Information Security (WEIS 2006), Robinson College, University of Cambridge, England. June 26-28, 2006, Pages 2-16.

[84] Adam Back, "Hashcash - A Denial of Service Counter-Measure", technical report, 1st August 2002.

[85] PineApp, “Recurrent Pattern Detection Technology”, RPD White Paper, January 2007.

[86] Commtouch, “ Commtouch – RPD™ Technology Network Based Protection Against Email-Borne Threats”, Commtouch Software Ltd. 2011, Page 3.

[87] David A. Wheeler, “Countering Spam by Using Ham Passwords (Email Passwords)”, 11th May 2011. Retrieved on 2011-11-11, Available from: http://www.dwheeler.com/essays/spam-email-password.html

[88] Marcelo H. P. C. Chaves. “Using Honeypots to Monitor Spam and Attack Trends”, ITU Regional Workshop on Frameworks for Cybersecurity and CIIP, Hanoi, Vietnam. August 2007.

[89] Nick Wallingford. “A Taste of Honey – UCE (Spam) Reduction Through Deception”. In S. Mann & T. Clear (Eds.), Proceedings of the Eighteenth Annual Conference of the National Advisory Committee on Computing Qualifications 2005, Pages 323-327.

[90] Mauro Andreolini Alessandro Bulgarelli Michele Colajanni Francesca Mazzoni, **“**HoneySpam: Honeypots fighting spam at the source”, SRUTI’05, Cambridge MA, USENIX Association, 7th July 2005, Pages 77-83.

[91] Kyumin Lee, James Caverlee, Steve Webb, *“The Social Honeypot Project: Protecting Online. Communities from Spammers*”,* Proceedings of the 19th international conference on World Wide Web 2010, Raleigh, North Carolina, USA, April 26-30, 2010, Publisher ACM New York, NY, USA ©2010, ISBN 978-1-60558-799-8, Pages 1139-1140.

[92] Greg, Mori, Malik, Jitendra. "Breaking a Visual CAPTCHA". Proceedings of the 2003 IEEE computer society conference on Computer vision and pattern recognition, Publisher: IEEE Computer Society Washington, DC, USA ,2003, ISBN:0-7695-1900-8 978-0-7695-1900-5, Pages 134-141.

[93] Captcha.net, "the Captcha Project", the official captcha site. Retrieved 2011-11-10, from: http://www.captcha.net/.

[94] John S Rhodes, "Opt In Email List Building: How to Build and Run a Successful Opt In List", ISBN-10: 1449500536, Publisher: CreateSpace, 24th September 2009,

[95] Tom M. Mitchell, "The Discipline of Machine Learning", School of Computer Science, Carnegie Mellon University,CMU-ML-06-108, July 2006.

[96] Richard O. Duda,Peter E.Hart, David G.Stork, "Pattern Classification (2nd Edition)", Publisher : Wiley-Interscience, ISBN 0471056693, October 2000.

[97] Øivind Due Trier, Anil K. Jain, Torfinn Taxt, "Feature Extraction Methodd for Character Recognition- A Survey", in "Pattern Recognition", Volume 29, Issue 4, April 1996, Pages 641–662

[98] U. Akilandeswari, R. Nithya, B. Santhi, "Review on Feature Extraction Methods in Pattern Classification", European Journal of Scientific Research, ISSN 1450-216X Vol.71 No.2 (2012), 2012, Pages 265-272.

[99] Mingqiang, Y., Kidiyo, K., Joseph, R., "A Survey of Shape Feature Extraction Techniques". Pattern Recognition Techniques, Technology and Applications, ISBN: 978-953-7619-24-4, November 1st 2008, DOI: 10.5772/6237.

[100] David Nguyen, Gokhan Memik, Seda Ogrenci Memik, and Alok Choudhary, "Real-Time Feature Extraction for High Speed Networks", 15th

International Conference on Field Programmable Logic and Applications, 2005, Pages 438-443.

[101] Anukool Lakhina, Mark Crovella, Christophe Diot, "Mining Anomalies Using Traffic Feature Distributions", Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, August22-26, 2005, Publisher: ACM New York, NY, USA, ISBN: 1-59593-009-4, Pages 217-228.

[102] W. Richard Stevens, "TCP/IP Illustrated, Volume 1: the Protocols", Addison-Wesley Professional, ISBN-10: 0201633469, 10th January 1994.

[103] RFC 793, Transmission Control Protocol, Information Sciences Institute, University of Southern California, September 1981.

[104] Steve Martin, Blaine Nelson, Anil Sewani, Karl Chen, Anthony Joseph, "Analyzing Behavioral Features for Email Classification", 2nd Conference on Email and Anti-Spam", CEAS 2005, at Stanford University, USA, July 26-27, 2005.

[105] Luiz Henrique Gomes, Cristiano Cazita, Jussara M. Almeida, Virg´ılio Almeida, Wagner Meira Jr, "Characterizing a Spam Traffic", in the proceeding of IMC' 04, Oct. 2004, Publisher: ACM New York, NY, USA, ISBN:1-58113-821-0, Pages 356-369.

[106] Cynthia Dhinakaran and Jae Kwang Lee, "Characterizing Spam traffic and Spammers", 2007 International Conference on Convergence Information Technology, Gyeongju , Source Korea, November 21-23, 2007, Publisher: IEEE Computer Society Washington, DC, USA, ISBN:0-7695-3038-9. Pages 831-836.

[107] Jung-Yoon Kim and Hyoung-Kee Choi, "Spam Traffic Characterization*", ITC-*CSCC2008 (23rd* annual conference), Japan, July 6-9, 2008.

[108] Barry Leiba, Joel Ossher, V.T. Rajan, Richard Segal, Mark Wegman, "SMTP Path Analysis", 2nd Conference on Email and Anti-Spam", CEAS 2005, at Stanford University, USA, July 26-27, 2005.

[109] Laura Bertolotti, Maria Carla Calzarossa, "Workload characterization of mail servers*", the proceedings of SPECT'2000, Vancouver Canada, July 16-20, 2000.

[110] Robert Beverly and Karen Sollins, "Exploiting Transport-Level Characteristics of Spam", 5th Conference on Spam and Anti-Spam, CEAS 2008, Microsoft Research Silicon Valley, Mountain View, California, August21-22, 2008.

[111] Luiz Henrique Gomes, Cristiano Cazita, Jussara M. Almeida_, Virg´ılio Almeida, Wagner Meira Jr, "Workload models of spam and legitimate e-mails", Performance Evaluation, Volume 64, Issues 7–8, Publisher: Elsevier Science Publishers B. V. Amsterdam, The Netherlands, The Netherlands ISSN: 0166-5316 August 2007, Pages 690–714.

[112] Fulu Li Mo-Han Hsieh Pawel Gburzynski, "The Community Behavior of Spammers", 2011. Available from: http://web.media.mit.edu/~fulu/ClusteringSpammers.pdf.

[113] M. L. Sang, S. K. Dong, and S. P. Jong, "Spam detection using feature selection and parameters optimization" in Proceedings of the 4th International Conference on Complex, Intelligent and Software Intensive Systems, (CISIS '10), Krakow ,Poland, February 2010, Publisher: IEEE Computer Society Washington, DC, USA, ISBN: 978-0-7695-3967-6, Pages 883-888.

[114] Prasanna Desikan and Jaideep Srivastava, "Analyzing Network Traffic to Detect E-Mail Spamming Machines", Proc. Workshop on Privacy and Security Aspects of Data Mining, Brighton, UK, 2004, Pages 67–76.

[115] Luiz Henrique Gomes, Cristiano Cazita, Jussara M. Almeida_, Virg´ılio

Almeida,Wagner Meira Jr. “Workload models of spam and legitimate e-mails”, Performance Evaluation-an international journal, Performance Evaluation 64, 2007,Pages 690–714

[116] Ni Zhang, Yu Jiang, Binxing Fang, Xueqi Cheng, Li Guo, “Traffic Classification-based Spam Filter”, IEEE International Conference on Communications, Istanbul, June 2006, Print ISBN: 1-4244-0355-3, Vol.5, Pages 2130-2135.

[117] Steve DiBenedetto, Kaustubh Gadkari, Nicholas Diel, Andrea Steiner, Dan Massey and Christos Papadopoulos, “Fingerprinting Custom Botnet Protocol Stacks”, 6th IEEE Workshop on Secure Network Protocols (NPSec), Koyot, October 2010, Print ISBN: 978-1-4244-8916-9, Pages 61-62.

[118] Alfredo H-S. Ang, Wilson H. Tang, “Probability Concepts in Engineering: Emphasis on Applications to Civil and Environmental Engineering” 2nd Edition,Publisher: Wiley, ISBN-10: 047172064X, Publication Date: February 13, 2006. Page 293-296.

[119] Eadie, W.T., D. Drijard, F.E. James, M. Roos and B. Sadoulet. “Statistical Methods in Experimental Physics”. Publisher: World Scientific. ISBN 981256795X, 9789812567956, 2006, Page 316.

[120] Lawrence Lapin and William D. Whisler, “Quantitative Decision Making” 7th edition, Table G. 24th September 2001. Publisher: South-Western College Pub, ISBN-10: 0534380247.

[121] Anti Spam Assistant, http://antispam-assistant-pro.com/index.asp.

[122] Kevin J. Connolly, “Law of Internet Security and Privacy”, Aspen Publishers Online, 2003, ISBN 0735542732, 9780735542730, Page 131.

[123] Lehigh University, “Assigning Weight Factor”, Weight Factor Handout. from:

http://www.lehigh.edu/~inhro/documents/GPS_WeightingFactors_Handout.pdf, Retrieved 2011-11-25.

[124] William J. Marshall and Stephen K. Bangert. “Clinical Biochemistry: Metabolic and Clinical Aspects”, Churchill Livingstone, June 20, 2008. ISBN-10: 0443101868, Page 12.

[125] Robin, “Machine Learning: An Overview”. 1st March 2010. http://intelligence.worldofcomputing.net/machine-learning/machine-learning-overview.html#, Retrieved 2011-11-11.

[126] Malathy K, “An Enhanced Fuzzy c-Means Clustering for Intelligent Prediction”, Proceedings of International Conference on Computing and Control Engineering, ICCCE 2012, Dr.M.G.R. Educational and Research Institute University, April 12-13, 2012, Paper ID: ICCCECS495.

[127] D.A.DeMillo, R.J.Lipton, and F.G.Sayward, “Hints on test data selection: Help for the practicing programmer”, Computer, volume: 11, Issue: 4, Publication Date: April 1978, Sponsored by: IEEE Computer Society ISSN: 0018-9162, Pages34-41.

[128] Kaspersky Lab. “Antispam, Evaluation Guide”, White Paper from Kaspersky Lab, 2011, Page 4.

Appendix 1: Thresholds for the Trigger System after Training Process (Percentile Value =95%)

1. Thresholds for Triggers in Time Intervals ($T_{current-th}[i]$):

Time Interval	Threshold for Trigger in time intervals	Time Interval	Threshold for Trigger in time intervals
T0(0:00~1:00)	232.431824	T12(12:00~13:00)	172.218185
T1(1:00~2:00)	230.831818	T13(13:00~14:00)	147.254547
T2(2:00~3:00)	238.668182	T14(14:00~15:00)	168.881821
T3(3:00~4:00)	156.240906	T15(15:00~16:00)	332.427277
T4(4:00~5:00)	156.240906	T16(16:00~17:00)	150.522720
T5(5:00~6:00)	156.240906	T17(17:00~18:00)	149.718185
T6(6:00~7:00)	161.418182	T18(18:00~19:00)	193.068176
T7(7:00~8:00)	123.381821	T19(19:00~20:00)	149.954544
T8(8:00~9:00)	178.509094	T20(20:00~21:00)	248.795456
T9(9:00~10:00)	261.186371	T21(21:00~22:00)	531.049988
T10(10:00~11:00)	295.463623	T22(22:00~23:00)	255.177277
T11(11:00~12:00)	302.059082	T23(23:00~24:00)	237.149994

2. Threshold for Trigger in a 24-hour Monitor Period ($T_{total-th}$) :

$T_{total-th}$= 5220.245605

Appendix 2: Thresholds and Weight Values for the Detection System after the Training Process (Percentile Value =95%)

1. Thresholds for 24 intervals ($TH_{Vol-Current}$):

Time Interval	$TH_{Vol-Current}$ (Number of Packets with Syn Flag set in each time interval)	Time Interval	$TH_{Vol-Current}$ (Number of Packets with Syn Flag set in each time interval)
TH0(0:00~1:00)	736.000000	TH12(12:00~13:00)	135.000000
TH1(1:00~2:00)	236.000000	TH13(13:00~14:00)	52.000000
TH2(2:00~3:00)	1253.000000	TH14(14:00~15:00)	48.000000
TH3(3:00~4:00)	743.000000	TH15(15:00~16:00)	283.000000
TH4(4:00~5:00)	363.000000	TH16(16:00~17:00)	735.000000
TH5(5:00~6:00)	645.000000	TH17(17:00~18:00)	463.000000
TH6(6:00~7:00)	462.000000	TH18(18:00~19:00)	747.000000
TH7(7:00~8:00)	352.000000	TH19(19:00~20:00)	268.000000
TH8(8:00~9:00)	219.000000	TH20(20:00~21:00)	835.000000
TH9(9:00~10:00)	120.000000	TH21(21:00~22:00)	346.000000

TH10(10:00~11:00)	81.000000	TH22(22:00~23:00)	2100.000000
TH11(11:00~12:00)	89.000000	TH23(23:00~24:00)	653.000000

2. **Threshold for total Packets with SYN in 24 hours:**

$TH_{Vol-total}$= 15909.000000

3. **Threshold for the coordinates of (FIN/SYN flag set, SYN):**

(0.112374, 13544.000000)

(0.431140, 15909.000000)

(0.356690, 17183.000000)

(0.276505, 19591.000000)

(0.375018, 20959.000000)

(0.540906, 21146.000000)

(0.384075, 23058.000000)

(0.154208, 27184.000000)

(0.251864, 32859.000000)

(0.419241, 36089.000000)

(0.274147, 41113.000000)

(0.399302, 50155.000000)

(0.490331, 50366.000000)

(0.418629, 52629.000000)

(0.183271, 52938.000000)

(0.400745, 60707.000000)

(0.339256, 72435.000000)

(0.447929, 76530.000000)

(0.215768, 91251.000000)

(0.360102, 135545.000000)

4. **Threshold for Number of Similar Payload:**

   $\boldsymbol{TH_{payload}}$= 1824.000000

5. **Threshold for OUT/IN Ratio:**

   $\boldsymbol{TH_{out-in}}$= 109.188797

6. **Threshold for Quiet Period Ratio:**

   $\boldsymbol{TH_{Time}}$= 0.328212

7. **Threshold for the final result D:**

   $\mathbf{D_{threshold}}$=2.062049

8. **Values of weights for Algorithms:**

- Weight for Algorithm 1 (Ratio of FIN/SYN Flag Set)

w1=0.888889

- Weight for Algorithm 2 (Relative to Time of Day)

  w2=0.684211

- Weight for Algorithm 3 (Similar Payload-size Connections)

  w3=0.666667

- Weight for Algorithm 4 (Volume of Connections in Current Interval)

  w4=0.555556

- Weight for Algorithm 5 (Volume of Connections in Recent 24 Hours)

  w5=0.642857

- Weight for Algorithm 6 (Ratio of OUT/IN SMTP Connections)

  w6=0.863636

Appendix 3: Results of Test Processes

Test Data 1:

1. **Percentile value=95% $D_{threshold}$=2.062049**

   Weight values:

   w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.863636

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	0	1	1	1	1	0	2.549290
2	1	1	1	1	1	1	4.301815
3	1	1	1	1	1	1	4.301815
4	0	1	1	1	1	1	3.412926
5	0	1	0	1	1	1	2.746260

2. **Percentile value=50% $D_{threshold}$=2.580796**

   Weight values:

   w1=0.888889, w2=0.600000,w3=1.000000,w4=0.730769,w5=1.000000,w6=0.850000

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	1	1	1	0	1	4.069658
2	0	0	1	1	1	0	2.730769
3	0	1	0	1	1	1	3.180769

Test Data 2:

1. **Percentile value=95% $D_{threshold}$=2.062049**

   Weight values:

   w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.863636

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	1	0	1	1	1	3.635149
2	1	1	1	1	1	1	4.301815
3	1	1	1	1	1	1	4.301815
4	1	1	0	1	1	1	3.635149
5	1	1	1	1	1	1	4.301815
6	1	1	1	1	1	1	4.301815
7	1	1	1	1	1	1	4.301815
8	1	1	1	1	1	1	4.301815
9	1	1	0	1	1	1	3.635149
10	**1**	**1**	**1**	**1**	**1**	**1**	4.301815
11	1	1	1	1	1	1	4.301815
12	1	1	0	1	1	1	3.635149
13	1	1	0	1	1	0	2.771512
14	0	1	1	1	1	1	3.412926
15	1	1	1	1	1	1	4.301815
16	0	1	0	1	0	1	2.103403
17	1	0	0	1	1	1	2.950938

2. Percentile value=50% $D_{threshold}$=2.580796

Weight values:

w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.863636

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	0	1	1	0	0	2.619658
2	1	1	0	1	1	1	4.069658
3	1	1	0	1	1	1	4.069658
4	1	1	0	1	0	1	3.069658
5	1	1	0	1	0	1	3.069658
6	1	1	1	1	0	1	4.069658
7	1	1	1	1	1	1	5.069658
8	1	0	0	1	1	1	3.469658
9	0	0	1	1	1	1	3.580769
10	1	1	1	1	0	1	4.069658

Test Data 3:

1. **Percentile value=95% $D_{threshold}$=2.062049**

   Weight values:

   w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.857143

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	0	1	1	1	1	1	3.412926
2	1	1	1	1	1	1	4.301815
3	0	1	1	1	1	1	3.412926
4	0	1	0	1	1	1	2.746260
5	1	0	1	1	1	1	3.617605
6	0	1	1	1	1	1	3.412926
7	1	1	1	1	1	1	4.301815
8	1	1	1	1	1	1	4.301815
9	0	1	1	1	1	1	3.412926
10	0	1	0	1	0	1	2.103403
11	1	1	1	1	1	1	4.301815
12	1	0	1	1	1	1	3.617605
13	0	1	1	1	0	1	2.770069
14	0	1	0	1	0	1	2.103403
15	0	1	1	1	1	1	3.412926
16	1	1	1	1	1	1	4.301815
17	0	1	1	1	1	1	3.412926
18	0	1	1	1	1	1	3.412926

19	0	1	0	1	1	1	2.746260
False Positive 1	0	1	0	1	0	1	2.103403

2. Percentile value=50% $D_{threshold}$=2.580796

Weight values:

w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.863636

Spam Relay	**Output of Algorithm 1**	**Output of Algorithm 2**	**Output of Algorithm 3**	**Output of Algorithm 4**	**Output of Algorithm 5**	**Output of Algorithm 6**	**Final Result (D)**
1	1	0	1	0	1	0	2.888889
2	1	0	1	1	1	1	4.469658
3	0	0	1	1	1	1	3.580769
4	1	1	0	1	0	1	3.069658
5	1	0	1	1	0	0	2.619658
6	1	0	0	1	1	1	3.469658
7	1	0	1	1	1	1	4.469658
8	1	0	1	1	0	0	2.619658
9	0	0	1	1	1	1	3.580769

Test Data 4:

1. **Percentile value=95% $D_{threshold}$=2.062049**

   Weight values:

   w1=0.888889, w2=0.684211, w3=0.666667, w4=0.555556, w5=0.642857, w6=0.863636

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	1	1	1	1	1	4.301815
2	1	1	1	1	1	1	4.301815
3	1	1	0	1	1	0	2.771512
4	1	1	1	1	1	1	4.301815
5	1	1	1	1	1	1	4.301815
6	1	0	1	1	1	1	3.617605
7	0	1	1	1	1	1	3.412926
8	1	1	1	1	1	1	4.301815
9	0	1	1	1	0	1	2.770069
10	**0**	**1**	**1**	**1**	**0**	**1**	2.770069
11	1	1	1	1	1	1	4.301815
12	1	1	1	1	0	1	3.658958
13	0	1	0	1	0	1	2.103403
14	0	1	0	1	0	1	2.103403
15	1	1	1	1	1	1	4.301815
16	1	1	0	1	1	1	3.635149
17	1	0	0	1	1	1	2.950938
18	0	1	0	1	1	1	2.746260

19	0	1	0	1	0	1	2.103403
20	1	0	0	1	1	0	2.087301
21	0	1	0	1	0	1	2.103403
22	0	1	1	1	1	1	3.412926
23	0	1	0	1	0	1	2.103403
24	1	0	1	1	1	1	3.617605
25	0	1	1	1	1	1	3.412926
26	0	1	0	1	1	1	2.746260
27	1	1	0	1	1	1	3.635149
28	1	1	0	1	1	1	3.635149
29	**0**	**1**	**0**	**1**	**0**	**1**	2.103403
30	0	1	1	1	1	1	3.412926
31	1	1	0	1	1	1	3.635149
32	0	1	0	1	1	1	2.746260
33	1	1	0	1	1	1	3.635149
34	0	1	0	1	0	1	2.103403
35	0	1	1	1	1	1	3.412926
36	0	1	0	1	1	1	2.746260
37	1	1	0	1	1	1	3.635149
38	0	1	0	1	1	1	2.746260
39	1	1	1	1	1	1	4.301815
40	1	1	0	1	1	1	3.635149
41	0	1	0	1	1	1	2.746260

2. Percentile value=50% $D_{threshold}$=2.580796

Weight values:

w1=0.888889, w2=0.600000,w3=1.000000,w4=0.730769,w5=1.000000,w6=0.850000

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	0	0	1	1	1	3.469658
2	1	0	0	1	1	1	3.469658
3	1	0	1	0	1	1	3.738889
4	1	0	1	1	0	1	3.469658
5	0	0	1	1	1	0	2.730769
6	1	0	1	1	1	1	4.469658
7	1	1	1	1	1	1	5.069658
8	1	0	0	1	1	1	3.469658
9	1	0	0	1	1	0	2.619658
10	1	0	1	1	1	1	4.469658
11	0	1	1	1	0	1	3.180769
12	1	0	0	1	1	1	3.469658
13	1	0	0	1	1	1	3.469658
14	1	0	0	1	1	1	3.469658
15	1	1	0	1	1	1	4.069658
16	1	1	0	1	1	1	4.069658
17	1	1	0	1	0	1	3.069658

Appendix 4: Results from System with Several Algorithms Enabled

1. Algorithm 1,3&5 Enabled

Test data set: Test Data Set 3

Percentile value=95% $D_{threshold}$=0.642857

Weight values:

w1=0.888889, w3=0.666667, w5=0.642857

Spam Relay	Output of Algorithm 1	Output of Algorithm 3	Output of Algorithm 5	Final Result (D)
1	0	1	1	1.309524
2	1	1	1	2.198413
3	0	1	1	1.309524
4	1	1	1	2.198413
5	0	1	1	1.309524
6	1	1	1	2.198413
7	1	1	1	2.198413
8	0	0	1	1.309524
9	1	1	1	2.198413
10	1	1	1	2.198413
11	0	1	0	0.666667
12	0	1	1	1.309524
13	1	1	1	2.198413
14	0	1	1	1.309524
15	0	1	1	1.309524
False Positive 1	1	0	0	0.888889
False Positive 2	1	0	0	0.888889

False Positive 3	1	0	0	0.888889

2. Algorithm1,3&6 Enabled

Test data set: Test Data Set 3

Percentile value=95% $D_{threshold}$=0.863636

Weight values:

w1=0.888889, w3=0.666667, w6=0.863636

Spam Relay	Output of Algorithm 1	Output of Algorithm 3	Output of Algorithm 6	Final Result (D)
1	0	1	1	1.530303
2	1	1	1	2.419192
3	0	1	1	1.530303
4	1	1	1	2.419192
5	0	1	1	1.530303
6	1	1	1	2.419192
7	1	1	1	2.419192
8	0	1	1	1.530303
9	1	1	1	2.419192
10	1	1	1	2.419192
11	0	1	1	1.530303
12	0	1	1	1.530303
13	1	1	1	2.419192
14	0	1	1	1.530303
15	0	1	1	1.530303
False Positive 1	1	0	0	2.103403
False Positive 2	1	0	0	2.103403
False Positive 3	1	0	0	2.103403

3. Algorithm3,4&5 Enabled

Test data set: Test Data Set 3

Percentile value=95% $D_{threshold}$=1.198431

Weight values:

w3=0.666667, w4=0.555556, w5=0.642857

Spam Relay	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Final Result (D)
1	1	1	1	1.865079
2	1	1	1	1.865079
3	1	1	1	1.865079
4	1	1	1	1.865079
5	1	1	1	1.865079
6	1	1	1	1.865079
7	1	1	1	1.865079
8	1	1	1	1.865079
9	1	1	1	1.865079
10	1	1	1	1.865079
11	1	1	0	1.222222
12	1	1	1	1.865079
13	1	1	1	1.865079
14	1	1	1	1.865079

15	1	1	1	1.865079

Appendix 5: Thresholds and Weight values after Update Process

Update Process was launched after the system was tested by using Test Data Set 3.

1. Thresholds for 24 intervals ($TH_{Vol-Current}$):

Time Interval	$TH_{Vol-Current}$ (Number of Packets with Syn Flag set in each time interval)	Time Interval	$TH_{Vol-Current}$ (Number of Packets with Syn Flag set in each time interval)
TH0(0:00~1:00)	464.000000	TH12(12:00~13:00)	89.000000
TH1(1:00~2:00)	258.000000	TH13(13:00~14:00)	78.000000
TH2(2:00~3:00)	735.000000	TH14(14:00~15:00)	48.000000
TH3(3:00~4:00)	324.000000	TH15(15:00~16:00)	48.000000
TH4(4:00~5:00)	363.000000	TH16(16:00~17:00)	83.000000
TH5(5:00~6:00)	274.000000	TH17(17:00~18:00)	151.000000
TH6(6:00~7:00)	462.000000	TH18(18:00~19:00)	364.000000
TH7(7:00~8:00)	253.000000	TH19(19:00~20:00)	253.000000
TH8(8:00~9:00)	73.000000	TH20(20:00~21:00)	424.000000
TH9(9:00~10:00)	119.000000	TH21(21:00~22:00)	177.000000
TH10(10:00~11:00)	80.000000	TH22(22:00~23:00)	356.000000
TH11(11:00~12:00)	81.000000	TH23(23:00~24:00)	356.000000

2. **Threshold for total Packets with SYN in 24 hours:**

$TH_{Vol-total}$= 12570.000000

3. **Threshold for the coordinates of (FIN/SYN flag set, SYN):**

(0.556841, 4926.000000)

(0.393600, 10625.000000)

(0.335800, 12570.000000)

(0.564087, 12803.000000)

(0.112374, 13544.000000)

(0.431140, 15909.000000)

(0.983560, 16788.000000)

(0.356690, 17183.000000)

(0.276505, 19591.000000)

(0.375018, 20959.000000)

(0.540906, 21146.000000)

(0.349058, 21707.000000)

(0.384075, 23058.000000)

(0.635677, 25299.000000)

(0.154208, 27184.000000)

(0.498839, 30146.000000)

(0.218232, 32053.000000)

(0.251864, 32859.000000)

(0.343009, 33314.000000)

(0.343009, 33314.000000)

(0.419241, 36089.000000)

(0.357647, 40512.000000)

(0.274147, 41113.000000)

(0.248324, 44309.000000)

(0.399302, 50155.000000)

(0.490331, 50366.000000)

(0.418629, 52629.000000)

(0.183271, 52938.000000)

(0.440544, 59742.000000)

(0.400745, 60707.000000)

(0.642655, 64190.000000)

(0.339256, 72435.000000)

(0.241232, 75587.000000)

(0.447929, 76530.000000)

(0.447929, 76530.000000)

(0.215768, 91251.000000)

(0.553942, 102731.000000)

(0.285980, 102731.000000)

(0.258148, 110646.000000)

(0.360102, 135545.000000)

4. **Threshold for number of Similar Payload:**

   $\boldsymbol{TH_{payload}}$= 1824.000000

5. **Threshold for OUT/IN ratio:**

   $\boldsymbol{TH_{out-in}}$= 109.188797

6. **Threshold for quiet period ratio:**

   $\boldsymbol{TH_{Time}}$= 0.354417

7. **Threshold for the final result D:**

   $\mathbf{D_{threshold}}$=2.950610

8. Values of weights for Algorithms:

- Weight for Algorithm 1 (Ratio of FIN/SYN Flag Set)

  w1=1.000000

- Weight for Algorithm 2 (Relative to Time of Day)

  w2=0.941176

- Weight for Algorithm 3 (Similar Payload-size Connections)

  w3=1.000000

- Weight for Algorithm 4 (Volume of Connections in Current Interval)

  w4=0.975610

- Weight for Algorithm 5 (Volume of Connections in Recent 24 Hours)

  w5=1.000000

- Weight for Algorithm 6 (Ratio of OUT/IN SMTP Connections)

  w6=0.975000

Appendix 6: Results of Test Processes on System after Update Process

Test Data 1:

1. **Percentile value=95%** $\mathbf{D_{threshold}}$**=2.950610**

   Weight values:

   w1=1.000000, w2=0.941176, w3=1.000000, w4=0.975610, w5=1.000000, w6=0.975000

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	0	1	1	1	1	0	3.916786
2	1	1	1	1	1	1	5.891786
3	1	1	1	1	1	1	5.891786
4	0	1	1	1	1	1	4.891786
5	0	1	0	1	1	1	3.891786

Test Data 2:

2. Percentile value=95% $D_{threshold}$=2.950610

Weight values:

w1=1.000000, w2=0.941176, w3=1.000000, w4=0.975610, w5=1.000000, w6=0.975000

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	1	1	0	1	1	1	4.891786
2	1	0	1	1	1	0	3.975610
3	0	1	1	1	1	1	4.891786
4	1	1	1	1	1	1	5.891786
5	1	1	0	1	1	1	4.891786
6	1	1	1	1	1	1	5.891786
7	1	1	1	1	1	1	5.891786
8	1	0	1	1	1	1	4.950610
9	1	1	1	1	1	1	5.891786
10	1	1	0	1	1	1	4.891786
11	**0**	**1**	**1**	**1**	**1**	**1**	4.891786
12	1	1	1	1	1	1	5.891786
13	1	1	0	1	1	1	4.891786
14	0	1	1	1	1	1	4.891786
15	1	1	1	1	1	1	5.891786
16	1	1	0	1	0	1	3.891786
17	1	0	0	1	1	1	3.950610

Test Data 4:

3. Percentile value=95% $D_{threshold}$=2.950610

Weight values:

w1=1.000000, w2=0.941176, w3=1.000000, w4=0.975610, w5=1.000000, w6=0.975000

Spam Relay	Output of Algorithm 1	Output of Algorithm 2	Output of Algorithm 3	Output of Algorithm 4	Output of Algorithm 5	Output of Algorithm 6	Final Result (D)
1	0	1	1	1	1	1	4.891786
2	1	1	1	1	1	1	5.891786
3	1	1	1	1	1	1	5.891786
4	1	1	1	1	1	1	5.891786
5	1	0	1	1	1	1	4.950610
6	0	1	1	1	1	1	4.891786
7	1	0	0	1	1	0	2.975610
8	1	1	1	1	0	1	5.891786
9	**1**	**1**	**1**	**1**	**0**	**1**	4.891786
10	1	1	0	1	1	0	3.916786
11	1	1	1	1	1	1	5.891786
12	1	1	1	1	1	1	5.891786
13	1	1	1	1	0	1	4.891786
14	1	1	0	1	0	1	3.891786
15	1	1	1	1	1	1	5.891786
16	1	1	0	1	1	1	4.891786
17	1	0	0	1	1	1	3.950610
18	0	1	0	1	1	1	3.891786
19	0	1	0	1	1	0	3.891786
20	1	0	0	1	1	0	2.975610
21	1	0	0	1	1	1	3.950610
22	1	1	0	1	0	1	3.891786
23	1	0	1	1	1	1	4.950610

24	1	1	0	1	1	1	4.891786
25	1	0	1	1	1	1	4.950610
26	0	1	1	1	1	1	4.891786
27	1	1	0	1	1	1	4.891786
28	**1**	**1**	**0**	**1**	**1**	**1**	4.891786
29	**1**	**1**	**0**	**1**	**1**	**1**	4.891786
30	1	1	0	1	0	1	3.891786
31	0	1	1	1	1	1	4.891786
32	1	1	0	1	1	1	4.891786
33	0	1	0	1	1	1	3.891786
34	0	1	0	1	1	1	3.891786
35	1	1	0	1	1	1	4.891786
36	0	1	0	1	1	1	4.891786
37	0	1	0	1	1	1	3.891786
38	0	1	0	1	1	1	3.891786
39	0	1	0	1	1	1	3.891786
40	1	1	1	1	1	1	5.891786
41	1	1	0	1	1	1	4.891786
42	0	1	0	1	1	1	3.891786
False Positive 1	0	0	1	1	1	0	2.975610