DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

SPAM FILTERING USING SENDER POLICY FRAMEWORK

by

Devrim SİPAHİ

February, 2017

İZMİR

SPAM FILTERING USING SENDER POLICY FRAMEWORK

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Computer Engineering

by
Devrim SİPAHİ

February, 2017
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"SPAM FILTERING USING SENDER POLICY FRAMEWORK"** completed by **DEVRİM SİPAHİ** under supervision of **ASST.PROF. DR. GÖKHAN DALKILIÇ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Asst. Prof. Dr. Gökhan DALKILIÇ

Supervisor

Prof. Dr. Yalçın ÇEBİ

Thesis Committee Member

Assoc. Prof. Dr. M. Süleyman ÜNLÜTÜRK

Thesis Committee Member

Prof. Dr. Yıldıray KILIÇASLAN

Examining Committee Member

Assoc. Prof. Dr. Murat Erten

Examining Committee Member

Prof.Dr. Emine İlknur CÖCEN

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

I would like to thank my thesis supervisor Asst. Prof. Dr. Gökhan DALKILIÇ, for his guidance, support and advice at every stage of this study. His valuable insights and experiences will guide me in all aspects of my future research.

I extend my thanks to the members of my committee, Prof. Dr. Yalçın ÇEBİ and Assoc. Prof. Dr. Mehmet Süleyman Ünlütürk for their useful comments and suggestions.

I would also thank to Asst. Prof. Dr. Zafer DİCLE and the employees of the Information Technology departments, for sharing her knowledge with me and for her support during the development of the applications.

In addition, I would like to thank the support from the Dokuz Eylul University Scientific Research Projects (Bilimsel Araştırma Projeleri, BAP) Coordination Unit. The number of the project was 2011.KB.FEN.31.

Finally, I thank my whole family. All of your support and love throughout my life have brought me to this achievement.

Devrim SİPAHİ

SPAM FILTERING USING SENDER POLICY FRAMEWORK

ABSTRACT

Spam has become one of the worst problems of email communication. Spam emails use Internet bandwidth redundantly. Spam emails cost people and companies extra money and time. Also, spam emails reduce the performance of networks and can lead to security problems.

Although various spam filtering methods have been developed, none of them is perfect. Creative spammers are capable of developing novel methods. Spam filtering can be grouped into content filtering and network-based filtering. Content filtering techniques require a lot of processor power. Therefore, network-based filtering methods, which require less resources, are gaining in spam-filtering. Sender Policy Framework (SPF) is a network-based filtering protocol to identify authorized servers which are allowed to send emails on behalf of a domain name. This way, spam mails can be filtered easily. But there are two deficiencies of the SPF implementation. Firstly, the SPF usage ratio is insufficient. Secondly, malicious usage is increasing.

As a solution to the first problem, we propose an estimation method called “Sender Authentication Network”. By analyzing the Internet Protocol (IP) addresses in A and MX records of the DNS system, their relation with the IP address in the SPF record is investigated. If there is no SPF record, a new one is created by predicting the classical DNS records A and MX. Since we have taken feedbacks from volunteers. Our method runs faster with low false-positive (FP) and false-negative (FN) values compared to the popular spam filtering methods.

To find a solution to the second problem, we have analyzed the term in the SPF records of the spam-sending domain names by comparing them with the legitimate domain names. We have developed a Naïve Bayes implementation the frequencies of SPF terms. This method helps to reduce the content-filtering processing time.

Keywords: Bayes Theorem, electronic mail, sender authentication, sender policy framework, spam filtering

GÖNDERİCİ POLİTİKASI ÇERÇEVESİ İLE İSTENMEYEN ELEKTRONİK POSTALARIN FİLTRELENMESİ

ÖZ

Spam postalar elektronik posta iletişimdeki en önemli sorunlardan biri haline gelmiştir. Spamlar gereksiz yere Internet trafiği oluşturur. Spam postalar insanların ve şirketlerin fazladan zaman ve para harcamalarına yol açar. Ayrıca spam postalar ağların performansını düşürür ve güvenlik sorununa yol açabilir.

Çeşitli spam filtreleme yöntemleri geliştirilmiş olmasına rağmen, bunların hiçbiri mükemmel değildir. Çünkü spam gönderenler yaratıcı olduklarından yeni yöntemler geliştirmektedirler. Spam filtreleme teknikleri, içerik filtreleme ve ağ filtreleme olarak iki gruba ayrılabilir. İçerik filtreleme teknikleri çok fazla işlemci gücüne ihtiyaç duyar. Bu nedenle, daha az kaynak gerektiren ağ tabanlı filtreleme yöntemleri, önem kazanmaktadır. Gönderici Politikası Çerçevesi (SPF) bir alan ismi için yetkili sunucuları tanıtan bir ağ tabanlı filtreleme protokolüdür. Bu sayede spam postalar kolaylıkla filtrelenebilir. Ama SPF protokolü uygulamalarının iki eksiği bulunmaktadır. Birincisi, SPF kullanım oranı yetersizdir; ikincisi kötü niyetli kullanımı artmaktadır.

İlk sorunun çözümü için "Gönderici Kimlik Doğrulama Ağı" adlı bir tahmin yöntemi önerdik. Bu yöntemde DNS sistemindeki A ve MX kayıtlarının IP adreslerini inceleyerek, SPF kaydındaki IP adresi ile ilişkisi incelenmiştir. Eğer SPF kaydı yoksa DNS sistemindeki A ve MX kayıtlarından tahmin edilerek yeni bir SPF kaydı oluşturulur. Kullanıcılardan aldığımız geri bildirimler, diğer popüler spam filtreleme yöntemleri ile karşılaştırıldığında, yöntemimiz düşük FP ve düşük FN değerleriyle hızlı çalışmaktadır.

İkinci sorunun çözümü için, spam gönderen alan isimlerinin SPF kayıtlarındaki terimlerle, normal eposta gönderen alan isimlerindeki terimleri karşılaştırdık. SPF

terimlerinin frekanslarını işleyen bir Yalın Bayes uygulaması geliştirdik. Bu yöntem içerik filtreleme işlem süresini azaltmaya yardımcı olmaktadır.

Anahtar kelimeler: Bayes Teorimi, elektronik posta, gönderici doğrulama, gönderici politikası çerçevesi, spam filtreleme

CONTENTS

Page

THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ vi
LIST OF FIGURES xii
LIST OF TABLES xiii

CHAPTER ONE – INTRODUCTION 1

1.1 Overview 1
1.2 Motivation of Spammers 2
1.3 The Cost of Spam 2
1.4 Anti-spam Techniques 3
1.5 Motivation 6
1.6 Aim of this Thesis 7
1.7 Thesis Organization 7

CHAPTER TWO – RELATED WORKS 9

2.1 Overview 9
2.2 Post-acceptance Methods 9
2.3 Pre-acceptance Methods 10
2.3.1 Whitelists/Blacklists 10
2.3.2 Greylisting 11
2.4 Sender Authentication Methods 11
2.4.1 Pretty Good Privacy (PGP) 12
2.4.2 Secure/Multipurpose Internet Mail Extensions (S/MIME) 12
2.4.3 Sender Policy Framework (SPF) 13
2.4.4 SenderID 13

2.4.5 DomainKeys Identified Mail (DKIM) 14
2.4.6 Author Domain Signing Practices (ADSP) 14
2.4.7 Domain-based Message Authentication, Reporting and Conformance (DMARC) 14
2.4.8 Vouch by Reference (VBR) 15
2.4.9 Summary 15

CHAPTER THREE – PROTOCOLS, PROGRAMS AND METHODS 17

3.1 Overview 17
3.2 Simple Mail Transfer Protocol (SMTP) 17
3.2.1 The SMTP Commands and Reply Codes and an Example 18
3.2.2 The Weaknesses of the SMTP protocol 20
3.3 The SPF Protocol 21
3.3.1 The History of the SPF Protocol 21
3.3.2 The Working Principle of the SPF Protocol 21
3.3.2.1 Publishing of SPF Record 22
3.3.2.1.1 The Mechanisms 22
3.3.2.1.2 The Modifiers 23
3.3.2.1.3 The Qualifiers 24
3.3.2.2 Checking Authorization of SPF Record 24
3.3.2.3 The Results of the Authorization Check 25
3.3.3 Global SPF Usage 26
3.3.4 The Deficiencies of the SPF Protocol 26
3.4 An Overview of the Qmail 27
3.5 Bayes' Theorem and Naïve Bayes Implementation 31
3.5.1 Naïve Bayes (NB) 33

CHAPTER FOUR – SPAM FILTERING USING SENDER AUTHENTICATION NETWORK 35

4.1 Overview 35

4.2 Sender Authentication Network 35
4.3 The Size of the Sender Authentication Network 36
4.3.1 The Comparison Method 37
4.3.1.1 The Dataset from deu.edu.tr Email Server 39
4.3.1.2 The Dataset from spf-all.com 40
4.3.2 The Results of Comparison 40
4.4 The Description of Sender Authentication Network 41

CHAPTER FIVE – IMPLEMENTATION OF THE SENDER AUTHENTICATION NETWORK 43

5.1 Overview 43
5.2 The Structure and Operation of the Method 43
5.3 The Results of the Implementation 46
5.3.1 The Statistics of the Results 46
5.3.2 The False-Positive Feedbacks and Whitelist 49
5.3.3 The Questionnaire Results 51
5.4 Comparison with SpamAssassin 55
5.5 The Weaknesses of the SAN 58
5.5.1 The Weaknesses That Cause False-Positives 58
5.5.2 The Weaknesses That Cause False-Negatives 58

CHAPTER SIX –DETECTING SPAMS THROUGH THEIR SPF RECORDS 60

6.1 Overview 60
6.2 The Required Materials and Methods 60
6.3 Preparation of Test Data 61
6.4 The Characteristics of the SPF Records 63
6.4.1 The Analysis of the Qualifiers 63
6.4.2 The Analysis of the Mechanisms and Modifiers 64
6.5 Implementation of Naïve Bayes on SPF Features 66

6.5.1 Appending Domain TLDs to the Method 68
6.6 Live Test and Comparison with SpamAssassin 69
6.7 Comparison with Other Methods like Exposure 71

CHAPTER SEVEN – CONCLUSION 72

7.1 Conclusion 72
7.2 Future Works 73

REFERENCES 74

LIST OF FIGURES

Page

Figure 3.1 The trip of email....18
Figure 3.2 The SMTP commands and replies codes 19
Figure 3.3 The functions of message transfer agent. 28
Figure 3.4 The structure of qmail and specific user identities. 29
Figure 3.5 The big picture of qmail system 31
Figure 5.1 The flowchart of the program. 45
Figure 5.2 The spam statistic from DEU and Symantec....48
Figure 5.3 The spam rates distribution per days of the week....49
Figure 5.4 The questionnaire results (a) daily FN number; (b) monthly FP number .54
Figure 6.1 Pseudo code of program. 66

LIST OF TABLES

Page

Table 3.1 The explanation of qualifiers. 24
Table 4.1 The comparison result prior to 2009 Marc. 38
Table 4.2 The comparison results. 39
Table 4.3 The detailed comparison results 41
Table 5.1 The spam statistics 47
Table 5.2 The SPF ratios per ham and spam emails. 48
Table 5.3 The false-positive information. 50
Table 5.4 Blocking ratios declerations. 52
Table 5.5 Number of false-positive declarations. 52
Table 5.6 Number of false-positive declarations. 53
Table 5.7 Summary of the first comparison of SAN and SpamAssassin. 56
Table 5.8 The comparison of SAN and SpamAssassin. 57
Table 6.1 The categorization of test data per studies 63
Table 6.2 The usage of prefixes of “all” in the first study 63
Table 6.3 The usage of prefixes of “all” in the second study 64
Table 6.4 The usage of SPF terms as complexity in the first study 65
Table 6.5 The usage of SPF terms in the second study 65
Table 6.6 The results of the implementation 67
Table 6.7 The distribution of TLDs per ham and spam groups 68
Table 6.8 The results of the implementation after appending TLDs 69
Table 6.9 Summary of the comparison with SpamAssassin 70

CHAPTER ONE
INTRODUCTION

1.1 Overview

Electronic mail (email) is the transmission of digital messages over computer networks such as the Internet. People exchange messages from and to anyone with an email address in the Internet. The current protocol for sending an email is the simple mail transfer protocol (SMTP) (RFC 822, 1982). It was proposed in 1982. This protocol is a simple protocol as its name indicates. At the time, there were a significantly small number of computer users compared to today. Therefore, there was no need for sender authentication.

With the growth of the Internet, this simplicity has turned into a weakness. Email sending has become an advertising area with an increasing number of email users. Advertising emails can easily be sent to millions of recipients with a little cost.

A spam email is defined as an unsolicited bulk email (UBE) (Spamhaus, 2016). An unsolicited email is one that is sent to a recipient without his/her consent. A bulk email is one sent to many users. A message is spam if and only if it is both bulk and unsolicited. First contact emails may be unsolicited, but not spam, and discussion lists emails are bulk, but not spam.

Gary Thuerk sent the first spam email in 1978. He was an aggressive marketer at Digital Equipment Corporation in those years. His spam email's content is a sales presentation of DEC 2020 computer. It was sent to 393 people on the ARPANET (Museum of Email, 2011) (EDN Network, 2015).

The spam rates began to increase in early 2000s. Until 2003, the spam rates were below 10%. In 2003, MessageLabs, which was purchased by Symantec in 2008, identified 40% of the scanned emails as spams. Since 2004 the ratio has increased to a level above 70%; it reached the highest rate in 2010 89% (Email Spam Trends at a

Glance, 2012). After 2011, the spam rates showed a decreasing trend. The overall spam trend for 2015 was around 52.7% (Symantec Intelligence Reports, 2016).

1.2 Motivation of Spammers

Sending email is one of the cheapest and most suitable communication activities. (Eklund, 2004). Users can easily send emails to many people. The size of an email message can vary in range from 1 kb to megabytes, which is larger than other communication devices like fax. Furthermore, anti-spam laws are not effective in all countries. The penalties for cyber-crimes are very high in USA & Europe. But, cyber laws in Asian countries like China, India, South Korea, where the number of Internet users are over a billion, are very different.

Spammers send a message to hundreds of millions of email accounts. If the positive response rate is higher than 0.001%, it is a good profit (Mangalindan, 2002). The cost of sending 100,000,000 emails is too low, and people rarely respond to sender, but with only 100 responses, spammers can make considerable profit.

1.3 The Cost of Spam

Spammers send 10,000,000 emails, to reach only 100 real recipients. The remaining 99,999,900 emails threat email communication. Spam emails consume valuable time and money as well as computational and bandwidth resources.

Spam wastes users' time. The average user spends 16 seconds reviewing and deleting each spam email, according to Nucleus Research (Edwards, 2016). Companies must use extra storage to quarantine spam emails due to the risk of making wrong decisions on whether an email is spam or not.

Spam emails lead to extra consumption of computing and network resources. They consume the bandwidth of networks, and extra CPU load of servers. Also, anti-spam solutions require extra computational resources. Spam emails cause security problems.

Computer viruses and trojans can be transferred by spam emails. Moreover, users' passwords can be stolen through spam emails.

According to the reports of Nucleus Research Inc., the average cost of spam per year per employee is $1,934 (Nucleus Research, 2015). This amount is more than twice the reported cost of the previous year.

1.4 Anti-spam Techniques

Various spam-filtering methods have been developed in order to solve the spam problem. A spam-filter should ideally have the following properties: Zero false positives (FP), zero false negatives (FN) and minimum cost. The term "FP email" refers to in reality a legitimate message mistakenly detected as spam; and FN email is a spam email which is not detected as spam. The costs of FP and FN are very different. FP is more important than FN. The cost of FN comes from working time for reading and deleting a spam message. But the cost of FP is much higher than FN, because there is a risk of missing opportunity due to delayed or lost emails.

Studies on spam filtering began in the late 1990s. In the following years, spam volumes increased rapidly. Therefore, various spam filtering approaches have been proposed. They can be grouped based on their stages of transmission, like pre-acceptance and post-acceptance (Twining et al., 2004). The pre-acceptance techniques are applied to the initial handshake, prior to message reception. They focus on blocking spam before the recipient's message transfer agent (MTA) server accepts emails; on the other hand, the post-acceptance techniques start after email has been received.

Some pre-acceptance methods are domain name system (DNS) Blacklists (DNSBLs) (Jung and Sit, 2004), DNS Whitelists (Levine, 2010) and greylisting (Gonzáles-Tavalán, 2006). DNS Blacklists record IP addresses of originated spam sources with using DNS queries. During a simple mail transfer protocol (SMTP) connection, the receiving MTA verifies the sending machine's IP address through querying its subscribed DNSBLs. If the sending IP address is listed in blacklist, the

email is rejected. Conversely, if sending IP address is listed in whitelist, the email is accepted without applying any other spam filters. Greylisting is a technique of challenging email servers to resend the email. Most spammers' software does not try to resend emails, if they have been rejected. But, legitimate email servers have to reattempt delivery for at least one day. Most spams can be blocked by this method. Other pre-acceptance methods are sender reputation (Sinha et al., 2008) and sender authentication (Herzberg, 2009).

Sender authentication methods are sender policy framework (SPF) (Wong and Schlitt, 2006), SenderID (Lyon and Wong, 2006) and DomainKeys Identified Mail (DKIM) (Allman et al., 2007). SPF and SenderID use DNS records of the domain name to authorize the sending IP address. SPF only checks the sender address given in the SMTP envelope called as return-path address, it does not care about the mail header. But, SenderID inspects the purported responsible address (PRA). The PRA can be one of the header fields like "From:", "Sender:", "Return-Path", or "Resent-Sender:". DKIM provides a digital signature for signing and verifying the email.

The post-acceptance methods use the content of the email message. These methods are classified as machine learning and non-machine learning techniques (Carpinter and Hunt, 2006). Non-machine learning methods are signatures, heuristics and traffic analysis. Heuristic analysis uses regular expression to detect common characteristics in spam messages. It applies a set of rules to the incoming email to detect these spam-like features (Gómez Hidalgo et al., 2000). The rules have to be updated frequently, because they are static; otherwise spammers can break these rules with new tricks.

Signature-based methods generate a unique hash value, called signature, for each known spam email (Damiani et al., 2004). Signature filters compare the signature of every incoming email with all the stored signatures of previously identified spam emails. This technique is ineffective for a new kind of spam. It requires more bandwidth.

Traffic analysis is used to find differences between spam traffic and non-spam

traffic (Gomes et al., 2004). Spam traffic differentiates some properties (anomalies) from normal traffic. These are; size of email, number of recipients per email, and locality of recipients. Anomalies and patterns in the Internet traffic can be detected by the log files of the SMTP servers at Internet service provider (ISP).

Machine learning algorithms are k-nearest neighbor (kNN), Support Vector Machine (SVM), Naïve Bayes, Markov Random Field (MRF) and Rocchio. Naive Bayes is a very popular spam filter. It is the first known email-filtering technique (Sahami et al., 1998). It works by correlating the use of words with spam and non-spam emails. It assumes that the words included in the email are independent from each other. Then, it uses Bayes' theorem to calculate probability of spam.

SVM is a supervised learning method used for classification and outliers' detection (Drucker et al., 1999). It uses decision planes which define decision boundaries. A decision plane separates unique class memberships. SVM uses labeled data, which is called supervised learning; then the algorithm outputs an optimal hyperplane which categorizes new examples. In essence, spam detection is a text classification. SVM is widely used in spam filtering.

KNN Analysis is a method to classify cases based on their similarity (distance) to other cases (Soucy and Mineau, 2001). It is a non-parametric and lazy learning algorithm. Non-parametric means that there are no assumptions on the underlying data distribution. Lazy learning means that there is no need to use the preliminary data points to do any generalization.

Rocchio algorithm is a method of relevance feedback, and was developed by using the Vector Space Model (Joachims, 1997). It creates a simple representation for each class (centroid). Classification is based on the similarity to distance from the centroid. It uses the word weighting method, then the document length normalization, and the similarity measure.

MRF is a graphical model of a common probability distribution, and it is similar to

a Bayesian network. However, contrary to Bayesian networks, which are directed and not cyclic, MRFs are undirected and may be cyclic. MRF is used to model the statistical actions of spam and non-spam (Chhabra et al., 2004).

There are also some products like SpamAssassin (SA) that combines different techniques. SA is a score based filter approach.

1.5 Motivation

An ideal spam filter should have the following features: no false positive (FP), no false negative (FN), minimum cost and working quickly. FP means that the email is in reality a non-spam (ham) message, wrongly detected as spam. FN means that the email is in fact a spam message that was not detected as spam. The cost of FP and FN are very different. The cost of FN is reading and deleting time of one spam message. But the cost of FP is much higher than FN, because an email can be lost. There is a risk of missing an important opportunity. Our goal is to filter out spam email by using minimum time and hardware. Our first priority is minimum FP emails. Undetectable spam can be filtered by a second filter.

Both of the pre-acceptance and post-acceptance methods have the advantages and weaknesses. The common advantages of pre-acceptance methods are that they run quickly and they require low computational resources. Conversely, the accuracy of spam detection is lower than post-acceptance methods.

We use pre-acceptance information (like source IP address and sender address) at the post-acceptance stage. All received emails are passed to quarantine before our program runs. Users can reach the emails in quarantine for further inspection. However, we use additional post-acceptance method for some users, who have more spam complaints.

Most of the pre-acceptance methods require initial data. Sender authentication methods can be initialized from scratch. The most commonly used method among

these methods is SPF (Taylor Bradley, 2006; WIDE Project, 2012). However, most system administrators do not use SPF. The SPF usage rate is only 27% (Stats, 2015). Therefore, additional methods should be developed for those who do not use SPF. On the other hand, spammers know SPF protocol. We have to detect malicious usage of the SPF protocol by spammers.

1.6 Aim of this Thesis

One of the purposes of this thesis is to predict SPF records of domains, which do not have SPF record. 72.7% of all domains do not have SPF records. But they have classical DNS records like A and MX. We will try to predict their SPF record through their classical DNS records.

One of the other intentions of this thesis is to detect malicious SPF records. Some spammers use SPF records with newly purchased domain names for sending spam. SPF records of these malicious domains are similar. They use same SPF terms to create SPF records. To detect these SPF records, we will analyze the statistical properties of SPF terms.

1.7 Thesis Organization

This thesis consists of seven chapters. In this chapter, we have stated the aim of the thesis, what we try to accomplish, and what our motivation is. The rest of the thesis is organized as follows.

Chapter 2 presents a literature survey on spam filtering methods. Also, related works about this thesis are introduced. Firstly, we classify spam filtering techniques. Then, we explain each technique with references.

Chapter 3 explains the protocols, programs and methods which are used in our method. Firstly, the protocols Simple Mail Transfer Protocol (SMTP) and Sender Policy Framework (SPF) are described. The deficiencies of SMTP and SPF protocol

are explained. Secondly, the qmail program is explained, which we have used as the Message Transfer Agent (MTA) server. Finally, Bayes theorem and Naïve Bayes classifier are explained.

Chapter 4 presents an introduction to our spam filtering method. We called it as Sender Authentication Network (SAN). Also the size of SAN is discussed. In this chapter, we describe dataset which we have used to specify the size of SAN. Finally, SAN is described with a real example.

In chapter 5, we begin to explain the implementation of the spam filtering system SAN. Firstly, the system architecture of the server is informed. Secondly, the details of the filtering system are explained. Thirdly, the results of the implementation and comparison with SpamAssassin are detailed. Finally, the weaknesses of the method are explained.

In chapter 6, our second method, detecting malicious SPF records through their SPF records, is explained. Firstly, the preparation of test data is explained. Secondly, the frequencies of SPF terms are calculated. Then, a Naïve Bayes implementation is described, using these frequencies. Finally, comparisons between live test, SpamAssassin and Exposure are made.

In chapter 7, we have presented our conclusions and the future direction of this thesis.

CHAPTER TWO
RELATED WORKS

2.1 Overview

In this chapter, related works about the thesis are introduced. Anti-spam solutions have been proposed as a response to spam emails that have been around since 1990s. The first spam filter was created by the owners of Listserv in 1995 (LISTSERV, 1995). The filter was created in response to the “green card” spam created one year earlier.

Email messages consist of a message header and a message body. The pre-acceptance methods focus on the message header; and the post-acceptance methods focus on both of these, but mostly use the message body.

2.2 Post-acceptance Methods

The email body is the main part of an email message. It contains the text of the message and other data. Therefore, spam filtering is one of the most interesting topics in the field of text classification. The first scholarly publication on text classification for spam filtering is by Sahami et al. (1998). They used Bayesian classification techniques to filter spam. Drucker et al. (1999) used support vector machines (SVM) for spam classification. They also compared it to three other classification algorithms: Rocchio, Ripper, and boosting decision trees. Carreras and Marquez (2001) used “boosting trees” for spam filtering. They showed that the Adaboost algorithm with confidence–rated predictions is a very well suited algorithm for spam filtering. Soonthornphisaj et al. (2002) proposed a centroid-based classification approach. They compared Naive Bayesian, k-nearest neighbor and centroid based approach. They found that the centroid-based approach is the most suitable method for spam filtering. O’Brien and Vogel (2003) made a comparison between Bayes and chi-squared. They suggested that both methods produced very promising results; but the chi-squared one had advantages with regard to reducing false positives. Cunningham and Delany (2007) presented an overview of techniques for k-Nearest Neighbor classification.

In particular, text classification techniques have gained success in spam filtering. However, spammers introduced a new trick, where the text-based spam message is embedded into images, which can make all current methods ineffective. Fumera et al. (2006) proposed an approach to filter spam, which exploits the text information embedded into images. They used Optical Character Recognition (OCR) tools to extract text from images.

2.3 Pre-acceptance Methods

The volume of spams has increased rapidly since 2004. Therefore, post-acceptance methods have become inadequate; whereas pre-acceptance methods gained importance. The pre-acceptance methods are unaware of the content of the email. Therefore, they use source IP address, sender address, message sending rate, and violation of SMTP standards. Because these methods are applied during SMTP sessions, they need to be implemented on the recipient's MTA servers.

2.3.1 Whitelists/Blacklists

A whitelist is a list, which includes all addresses (email address, domain name or IP addresses) from which we always accept to receive mails. We can add email addresses or domain names or IP addresses to this list. A blacklist is a list of addresses from which we do not want to receive emails. One of the oldest methods of spam filtering is DNS blacklisting. It uses a centralized database to block emails from hosts being used to send spams. The databases are updated by adding entries for hosts, which are used by spammers (RFC, 2010). During an SMTP session, an email server performs a DNS query. According to information received by the query, the email server makes a decision; as to whether accept or the incoming message. This method has two advantages; it depends on system resources less and it is easier to maintain. To use this method, the email server needs extra DNS query. Despite its advantages and ease of use, this method has a number of serious flaws. For instance, spammers can hide behind the anonymity of large ISP. Legitimate sites can be blacklisted either

incorrectly or because the source of a spam message appears to originate from the site. It is usually not easy for an administrator to remove a site from the blacklist database. Another disadvantage is the dependency on external sites. If a problem occurs when accessing the DNS server, this method is unusable.

2.3.2 Greylisting

Greylisting is designed to detect spams from poorly written software for email sending servers. This method rejects emails on the assumption that legitimate MTAs will always retry, while spammers' MTA software will not retry; because spammers send emails to millions of recipients. It is not a problem in terms of the spammer, that a few recipients of them did not receive spam emails. Greylisting works by returning a temporary failure response to the first attempt of sender MTA, but accepting it on the second attempt. This method looks at three pieces (called triplet) of information. These are the IP address of sender, the envelope sender address (return-path) and envelope recipient address. If this triplet has never been seen before, then the system rejects email within a certain period of time with a temporary failure. This email is accepted at the second attempt with known triplet. The main advantage is that greylisting needs minimal configuration. The second benefit is that rejecting email with a "temporary 451 error" is very cheap with regard to system resources. During the initial testing of greylisting in mid-2003, spam blocking rate was over 95%, and there was no legitimate email permanently blocked at all (Harris, 2003). One of the disadvantages of greylisting is the delay of the message. This method increases network traffic. Moreover, spammers can learn this method, and will develop new techniques to bypass it.

2.4 Sender Authentication Methods

Pre-acceptance methods also include sender authentication. Lists-based methods require preliminary information. Namely, a whitelist or blacklist needs a list of domain names or IP addresses. However, sender authentication techniques can be installed from scratch. Sender Authentication methods are Pretty Good Privacy (PGP),

Secure/Multipurpose Internet Mail Extensions (S/MIME), Sender Policy Framework (SPF), SenderID, DomainKeys Identified Mail (DKIM), Author Domain Signing Practices (ADSP), Domain-based Message Authentication, Reporting and Conformance (DMARC), and Vouch by Reference (VBR).

2.4.1 Pretty Good Privacy (PGP)

PGP is computer software used to encrypt and decrypt data (files, texts, e-mails, directories) over the Internet. PGP is often used in emails in order to increase the security of email communications. PGP was developed by Philip R. Zimmermann in 1991 (Zimmermann, 1991) and has become a standard way of providing email security. PGP was standardized by the name of OpenPGP standard in 2007 for encrypting and decrypting data (RFC 4880, 2007). PGP uses public-key cryptography which uses pairs of public and private keys. The private key is known only by the owner, and the public key is known by everyone. If the message is encrypted by the private key, only the public key can decrypt it. If the message is encrypted by the public key, only the private key can decrypt it.

2.4.2 Secure/Multipurpose Internet Mail Extensions (S/MIME)

S/MIME is used for sending digitally signed and encrypted messages. It was developed by RSA Data Security. The MIME is the standard format for extended Internet electronic mail (RFC 2045, 1996). It supports text in character sets, non-text attachments (audio, video, image, and program), message bodies with multiple parts, and non-ASCII character set in the header. S/MIME is a protocol which adds digital signatures and encryption to the MIME protocol. The last standards of S/MIME protocol are described in RFC 5751 (RFC 5751, 2010). It provides privacy, authentication, message integrity, non-repudiation of origin, and data security (using encryption).

2.4.3 Sender Policy Framework (SPF)

SPF is a standard that defines a method to prevent sender address forgery. SPF protocol was proposed by Wong and Schlitt in 2006. This protocol is described in RFC 4408 (RFC 4408, 2006). In this method, the receiving email servers check that whether an incoming email from a domain comes from an authorized host (IP) by that domain's administrators or not. SPF protocol allows a domain, to specify which IP addresses are authorized to use for its outgoing email servers. It uses Domain Name System (DNS) record. The SPF record is added to the DNS zone with a TXT record and it specifies authorized SMTP servers for the domain. An SPF record begins with the version string "v=spf1". SPF prevents domain name forgery of email addresses.

2.4.4 SenderID

Sender ID is an email sender authentication protocol proposed by Microsoft to provide protection against domain spoofing and phishing. This protocol was described in RFC 4406 (RFC 4406, 2006). The Sender ID Framework is based on SPF, with just a few changes. SPF only verifies the "Return-path" address, also known as the envelope sender. Sender ID verifies sender address identity based on the Purported Responsible Address (PRA). The PRA for most email system will be "From:" header field, the "Sender:" header field. However, the PRA is based on "Resent-*" header fields that are often not displayed to the user. The SenderID record is added to the DNS zone with a TXT record and it identifies authorized SMTP servers for the domain. Sender ID is almost identical to SPF except the initial string. Sender ID records can begin with "spf2.0/pra", "spf2.0/mfrom", or "spf2.0/mfrom,pra". mfrom is equal to SPF. "mfrom,pra" means verifying both the envelope sender and the PRA.

2.4.5 DomainKeys Identified Mail (DKIM)

DKIM is an email authentication protocol that allows an organization to take responsibility for a message that is in transit. It can be verified by a recipient. This protocol was developed by Yahoo, and is described in RFC 5585 (RFC 5585, 2009). It is defined to be a domain-based digital signature authentication framework for email, using public-key cryptography. There are two distinct operations: signing and verifying. The signing is made by the sender side. The signing module is inserted to the email header like “DKIM-Signature: v=1; a=rsa …”. Verification is performed using the signer's public key published in the DNS system. The DKIM records in the DNS are like “v=DKIM1; k=rsa; …”. DKIM is based on domain names, and signing is controlled by the domain administrator.

2.4.6 Author Domain Signing Practices (ADSP)

ADSP is an extension to DKIM. It provides recommendation to the validating email receiver about what to do if an email item is not signed. This protocol is described in RFC 5617 (RFC 5617, 2009). DKIM usage ratio is very low. The absence of a DKIM signature on an email does not mean spam. Therefore, some domains want to decide to sign all of their outgoing email. ADSP allows for a mechanism to signal the receiving side when a DKIM signature should be expected. The ADSP record in DNS is like: “_adsp._domainkey IN TXT dkim=discardable;".

2.4.7 Domain-based Message Authentication, Reporting and Conformance (DMARC)

DMARC is an email authentication protocol, which was built on the widely deployed SPF and DKIM protocols. It allows a receiving organization to check whether an incoming email from a domain is authorized and has not been changed during transport. It provides a procedure for the email receiver to report back to the sender's domain about emails that pass/fail DMARC evaluation. This protocol is described in RFC 7489 (RFC 7489, 2015). DMARC records are published in DNS

system with a subdomain label _dmarc (.demarc.domain.com). The content of the TXT record is like: "v=DMARC1; p=none; sp=quarantine; pct=100; rua=mailto:reports@domain.com".

2.4.8 Vouch by Reference (VBR)

VBR is a protocol for adding third-party sender certification to the Internet email system. It provides independent third parties to certify the domain owner which is associated with the received email. VBR information can be used by message transfer agent (MTA), mail user agent (MUA) or mail delivery agent (MDA). This protocol is described in RFC 5518 (RFC 5518, 2009). The first part requires the sender to add information to the email header. It creates a field which is called VBR-Info. The information enables the receiver to have access to senders' domain, the message type and a domain list that the sender expects to vouch for the legitimacy of the email. The VBR-Info field format is as follows:

"VBR-Info: md=<domain>; mc=<type-string>; mv=<certifier-list>".

When a receiver wants to verify whether a certification claim is valid or not, it compares the messages list to the services list through DNS. The receiver will query a TXT record for the name "domain.name.example._vouch.vouching.example".

2.4.9 Summary

The sender authentication methods allow receivers to properly identify the sender, so it can make quicker decisions about the delivery of the email. The spammers usually send malicious email with using trusted domain names. Therefore, sender authentication has become a good practice for senders. Amir Herzberg describes and compares the following DNS-based sender authentication mechanisms: SPF, DKIM and Sender-ID (Herzberg, 2009). The limitations of these mechanisms were clarified, and the email senders are advised to publish both SPF and DKIM policies to prevent email forgery.

We started this project in 2007. At the time, the SPF protocol was the most commonly used protocol all of the sender authentication protocols. Therefore, we preferred SPF protocol.

CHAPTER THREE
PROTOCOLS, PROGRAMS AND METHODS

3.1 Overview

This chapter explains the protocols which are used in our method. These are Simple Mail Transfer Protocol (SMTP) and Sender Policy Framework (SPF). The pros and cons of these protocols are examined. The qmail is used as an MTA server. A brief description of the usage of qmail is given. Bayes' theorem and naïve bayes implementation, which are used in this method, are explained theoretically. Database design is also handled in this section.

3.2 Simple Mail Transfer Protocol (SMTP)

SMTP is an Internet standard for sending and receiving emails. This protocol was defined by RFC 821 for the first time in 1982. The last version of SMTP protocol (with a new name Extended SMTP) was defined by RFC 5321 in 2008. SMTP is used to transmit emails by email servers and other message transfer agents (MTA). Client application, called Mail User Agent (MUA), uses SMTP to send emails to an email server for relaying. To receive emails, they use the Post Office Protocol (POP3), which uses TCP port 110, or the Internet Message Access Protocol (IMAP), which uses TCP port 143. Client applications access their email accounts through an email server to receive email messages. SMTP uses TCP port 25. Figure 3.11 shows the email communication from sender to receiver.

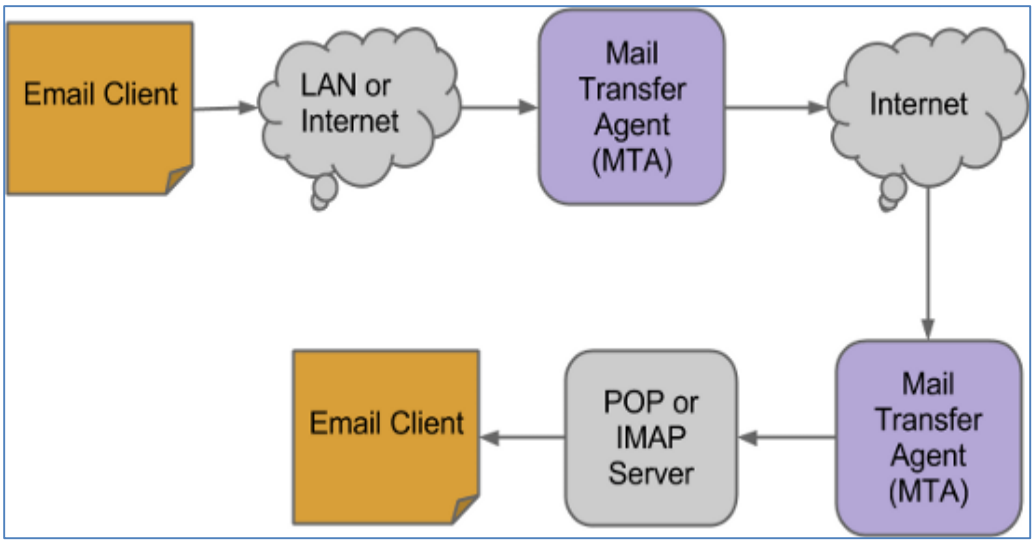


Figure 3.1 The trip of email (Arora, 2013).

An email client program like outlook, thunderbird or webmail application is used to create an email. This email goes to the MTA server to which the email client is connected. The client program uses TCP port 25 or 587. This server looks out for the recipient's address. The address is of the form "name@domain.com". The server uses the DNS technique to resolve the domain name into an IP address. Then it sends the email to the server, which has this IP address, over the Internet. Now, the email transmits over the Internet and reaches the destination MTA server. This server stores all the emails and these emails are accessible to email clients through POP3 or IMAP services.

3.2.1 The SMTP Commands and Reply Codes and an Example

To begin SMTP session, the sender establishes a TCP connection to the SMTP receiver. Then, the receiver sends a ready message (220) with its hostname and protocol name ESMTP. The sender sends a HELO command and the receiver replies with the reply code 250, which means that requested email action is completed. Then, the sender sends "mail from:" command to identify the sender name. The receiver sends an okay message (250). The sender sends "rcpt to:" command to identify the receiver address. The message transmission begins with "data" command, and ends with only one dot "." sign. Figure 3.2 shows the SMTP commands and reply codes.

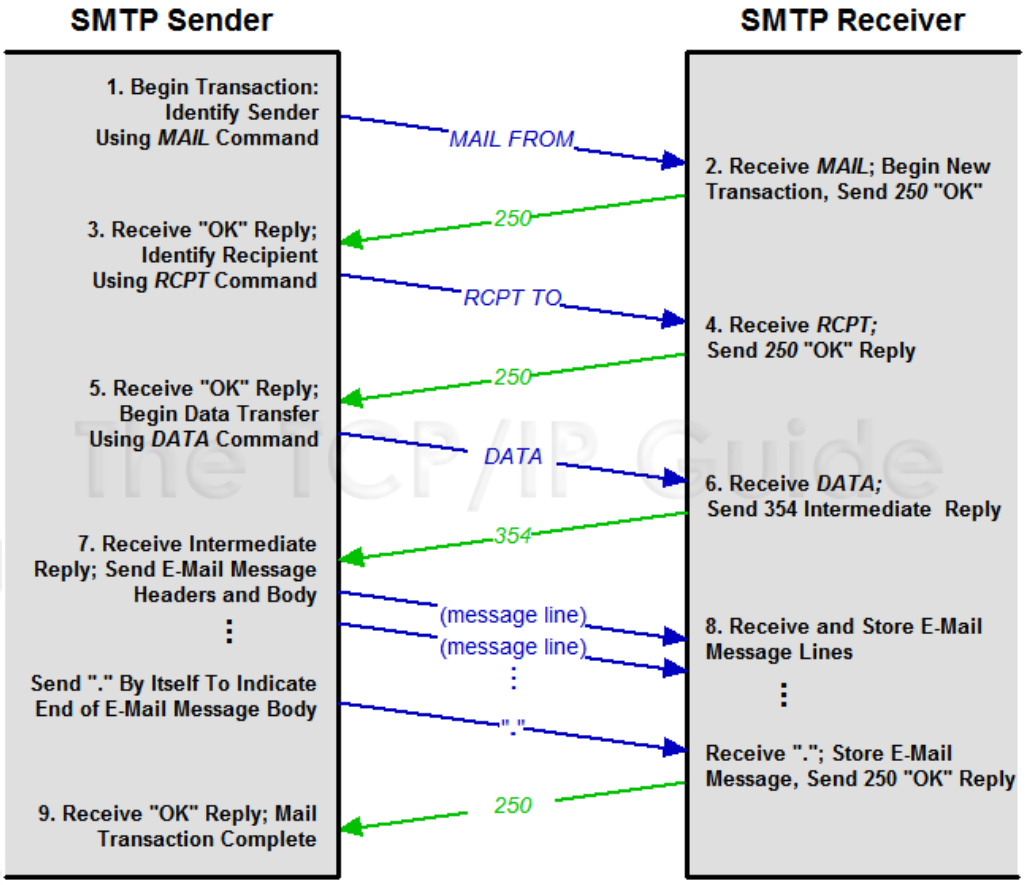


Figure 3.2 The SMTP commands and replies codes (SMTP Mail Transaction Process, 2005).

In this example, we used our main email server. We typed in italic, and server responses are shown in normal type.

```
% telnet mail.deu.edu.tr 25
Trying 193.140.151.7...
Connected to mail.deu.edu.tr.
Escape character is '^]'.
220 kordon.adm.deu.edu.tr ESMTP
helo
250 kordon.adm.deu.edu.tr
mail from:devrim@xxx.yy.zz
250 ok
rcpt to:devrim.sipahi@deu.edu.tr
250 ok
Data    //starting message
354 go ahead
From: XXXXX
```

```
To: YYYY
Subject: ZZZZZ

The message is starting.
….
The message is finished.
.   /end of messages
250 ok 1451303548 qp 166
quit / The session is finished.

221 kordon.adm.deu.edu.tr
Connection closed by foreign host.
```

3.2.2 The Weaknesses of the SMTP protocol

The SMTP protocol was written in 1982. At the time, there were fewer computers, fewer users and the Internet speed was significantly slower compared to how it is today. It was easy to know who is using which IP address, due to the small number of users. Therefore, there was no need to verify the sender address (sender authentication). Also, it was impossible to send bulk email, due to the small number of recipients and low speed of Internet.

Today, the number of Internet users has increased over third billion, Internet speed has risen up, and speed and capacity of computers are much more developed. This protocol has become inadequate in the face of such rapid growth. In the example above, users can write whatever they want on the field “mail from”. There is no control over this field. Therefore, any host (email server) can identify itself as any domain name it wants. This weakness is exploited by spammers by using fraud emails. To trace an email back to its source is difficult. Also, spammers can hide their identity to avoid obligation.

3.3 The SPF Protocol

Sender Policy Framework (SPF) is a domain–based sender authentication approach, with a checking system to allow receivers to check whether an incoming email from a domain comes from an authorized host by domain's administrators. Thereby, spammers who usually disguise the origin of the email are discouraged from email spoofing. The list of authorized hosts is published in the Domain Name System (DNS) through TXT records.

3.3.1 The History of the SPF Protocol

Any host can identify itself with any domain name it wants using SMTP. To solve this problem, Jim Miller suggested outbound-smtp mx DNS records to verify sender address in 1997. Bill Cole suggested “Mail Sender” (MS) DNS records to specify outgoing email servers of a domain in 2000. David Green published "Mail-Transmitter RR" on namedroppers’ mail list in 2002. Paul Vixie sent to namedroppers’ mail list a draft which is called "Repudiating MAIL FROM". Hadmut Danisch (2002) published as Internet draft, the first version of Reverse MX (RMX), which is used for SMTP sender authentication. Gordon Fecyk (2003) published an Internet draft, "Designated Mailers Protocol" (DMP). Meng Weng Wong combined the RMX and DMP specifications in 2003, and posted the first version of "Sender Permitted From" as a first message on a new mail list. SPF was firstly meant “Sender Permitted From” and was also known as SMTP+SPF. This name was determined as “Sender Policy Framework” in February 2004. The Internet Engineering Task Force (IETF) created the “MTA Authorization Records In DNS” (MARID) working group. This group use SPF and Microsoft's CallerID proposal, also known as Sender ID. The SPF community departed from MARID, and returned to the original version of SPF, and they published SPF RFC as an experimental RFC 4408, in 2006.

3.3.2 The Working Principle of the SPF Protocol

The operation of this protocol takes place in two stages. Firstly, the sender publishes

an authorized hosts list. This description is represented in an SPF record, which is published in DNS records. Secondly, the receiver performs a SPF check for each email message. An SPF client program checks the SPF record, in order to determine whether an email comes from an authorized source.

3.3.2.1 Publishing of SPF Record

The publishing of the SPF record is made by adding extra records to its existing DNS information. Domain names usually have A, PTR, and MX records in the DNS system. SPF record uses TXT field, which is a type of resource record in the DNS. SPF records have to start with the version string “v=spf1”. Then mechanisms (with qualifiers) or modifiers are written. If modifiers are not used, SPF records should end with the “all” mechanism. An example SPF record is given below:

domain.com. TXT "v=spf1 ip4:1.2.3.0/24 -all”

The record above states that the IP addresses which are in the 1.2.3.0/24 network are authenticated to send emails with the name of “example.com”. Emails are accepted if sent from this network, but rejected otherwise. The SPF records consist of the SPF version number, mechanisms, qualifiers (prefixes), and modifiers.

3.3.2.1.1 The Mechanisms. There are eight mechanisms; all, ip4, ip6, a, mx, ptr, include, exist.

• **all:** This mechanism is a check which always matches. It usually goes at the end of the SPF record. It is used to define all others. The prefix “-“ is recommended.

• **ip4:** The "ip4:" mechanism specifies an IPv4 address or network range. It can be used in the following form: ip4:<ip4-address> or ip4:<ip4-network>/<prefix-length>.

• **ip6:** The "ip6:" mechanism specifies an IPv6 address or network range. It can be the following form: ip6:<ip6-address> or ip6:<ip6-network>/<ip6 prefix-length>.

• **a:** The “a” mechanism is like “A” record in the DNS system. It specifies the IP address of domain, unless a prefix-length is given. It can be in the following form: a or a/<prefix-length> or a:<domain> or a:<domain>/<prefix-length>.

• **mx:** The “mx” mechanism is like “MX” record in the DNS system. All of the “A” records for all the “MX” records for domain are tested. It can be in the following form: mx or mx/<prefix-length> or mx:<domain> or mx:<domain>/<prefix-length>.

• **ptr:** The “ptr” mechanism is like “PTR” record in the DNS system. It tests the existence of the DNS reverse-mapping for the sending IP address. It is validated: at least one of the A records matches the original client IP. It can be in the following form: ptr or ptr:<domain>.

• **include:** The “include” mechanism starts a recursive evaluation. The specified domain is searched for a match. It has to be the following form: include:domain.

• **exists:** This mechanism tests the existence of a domain, namely tests a DNS A record. It allows for complex schemes involving optional domain name of the email envelope to specify what is allowed.

3.3.2.1.2 The Modifiers. The modifiers provide additional information. There are two modifiers: redirect and exp. Modifiers are optional. A modifier can appear only once in the SPF record, but must appear at the end, after all mechanisms.

• **redirect:** The SPF record for a domain replaces the current record. It has to be in the following form: redirect=<domain>. It is used instead of all the mechanism to refer to the policy record of another domain.

• **exp:** It means explanation string. When an MTA receiver rejects a message, it can respond with an explanation. It has to be in the following form: exp=<domain-specs>.

3.3.2.1.3 The Qualifiers. The qualifiers are located in front of the mechanisms as prefixes. The qualifiers are “+” (PASS), “?” (NEUTRAL), “~” (SOFTFAIL), “-“ (FAIL). Plus “+” is used for a PASS result. This can usually be omitted. The question mark “?” is used for a NEUTRAL result evaluated like NONE. It means there is no policy. The tilde symbol “~” is used for SOFTFAIL, a debugging assistance between FAIL and NEUTRAL. A SOFTFAIL respond means, that the message is accepted but labeled. The minus sign “-“ is used for FAIL, the email has to be rejected. The qualifiers are used as prefixes in the mechanism “ALL” as a decision. The qualifiers are shown in Table 1.

Table 3.1 The explanation of qualifiers.

Qualifier	**Result**	**Explanation**
+	Pass	The address passed the test; accept the message
-	Fail	The address failed the test; reject the message
~	Softfail	The address failed the test, but the result is not absolute; accept the message but mark it
?	Neutral	The address did not pass or fail the test; accept the message

3.3.2.2 Checking Authorization of SPF Record

An email client (receiver) implements a series of SPF tests for each email. It uses ordinary DNS queries, and then interprets the SPF information and acts upon the result. Firstly, the “envelop-sender address” (mail from) identity has to be checked, but the “helo” identity should also be checked just in case. To do this, the client (receiver) has to evaluate the “check_host()” function by using the arguments “IP”, “sender” and “domain”. The “IP” is the IP address of the sender (SMTP client). The “sender” is the "mail from", “Return-Path” or "helo" hostname identity. The “domain” is the domain part of the "mail from" or "helo" identity.

3.3.2.3 The Results of the Authorization Check

The result is not only pass or fail. The checking function may stop working due to error. There are seven possible query results, including pass, fail, none, neutral, softfail, temperror, permerror.

Pass: It means that the sender IP address is authorized to send emails with the given address.

Fail: This result is an absolute expression that the sender host (IP address) is not authorized to use the domain name in the given address.

None: This result means that there is no published record for the domain name.

Neutral: This result means that the domain administrator has clearly stated that it is not specified whether the IP address is authorized or not. It is similar to the "None" result. The only difference is that the “Neutral” is a deliberate notice of the domain administrator.

Softfail: This result is interpreted between "Neutral" and "Fail". The domain owner believes that the sender IP is not authorized, but it is not for sure. Therefore, the receiving program should not reject the email.

Temperror: This result means that a temporary error is encountered during the SPF check. The receiving program can either accept or reject the email temporarily.

Permerror: This result means that the SPF records could not be correctly evaluated. This indicates that an error has occurred, which has to be resolved by the DNS administrator.

3.3.3 Global SPF Usage

Although the SPF record syntax is very detailed, it is easy to use. Auxiliary tools for creating SPF records can be easily obtained from the Internet. For example, the www.openspf.org contains useful information and discussion. Most of spammers also use the following domain names as a sender address: gmail.com, Hotmail.com, live.com, yahoo.com. Therefore, the popular email sites like Gmail, Hotmail and Yahoo use SPF. Gmail, Hotmail and Yahoo recommend their users to create a Sender Policy Framework (SPF) record for their domain. The email traffic of these companies is high. An SPF check is the simplest, cheapest and quick method to prevent fraud addresses. With the most popular email providers' support, more than half of the emails coming from the domains contain an SPF record (Bradley, 2006). But, in general the SPF utilization rate is 27.3% (spf-all.com, 2016).

3.3.4 The Deficiencies of the SPF Protocol

The SPF protocol is useless, if the sender does not publish an SPF record. It is quite simple to create and publish an SPF record in the DNS system; but the usage ratio of the published SPF record is not sufficient. As mentioned in the previous section, the ratio of domain names, which do not use SPF protocol, is higher. The second problem of the SPF protocol is that most of the policies are useless. These are "?all" (neutral), "~all" (softfail) and the default policy "+all" (pass). The only policy that works is the "-all" (fail) policy, because the receiving email server rejects the email only with this policy. Therefore, it is recommended that the domain owners use the "-all" policy in their SPF records (Ostrihon and Rajabiun, 2008). In the site "spf-all.com", the ratios of the qualifiers are shown. The ratios are 31.3% for "-all", 33.9% for "~all", 34.2% for "?all" and 0.6% for "+all". The ratio 31.3 actually means 8.55% of over 88 million domains. In our dataset, the ratio is 24%.

To solve this problem, we proposed a method called Sender Authentication Network. This method predicts the SPF record of domains, which do not have any SPF record. This method will be described in the next section.

Another problem with this protocol is that, the spammers purchase new domain names and use the SPF records of these domain names to send spam emails. Therefore, we have to detect the malicious SPF records. Previous studies show that, the only method to prevent such spam would be using the blacklist or content filtering methods. To solve this problem, we proposed a new method, which analyzed the terms in the SPF record of malicious and legitimated domain names. The name of this method is "Detecting spam through their Sender Policy Framework records". This method will be described in the upcoming sections.

3.4 An Overview of the Qmail

The qmail is a Message Transfer Agent (MTA) package for the Unix environment. It was written by Daniel Julius Bernstein after 3 years of work in 1998. It was designed as an alternative to Sendmail for security related reasons. Security is the main reason for the development of qmail. It is designed structurally. This design makes qmail not only secure but also easy to understand and maintain.

The qmail software package has had a security guarantee since 1997. Daniel J. Bernstein promised to offer $500 to the first person, who finds a verifiable security hole in the qmail package (Bernstein, 2007).

The qmail is an MTA. The most popular MTA is Sendmail. MTA is a program which transmits electronic mail messages between computers. An ideal MTA must be reliable, in the sense that it never loses a message. An ideal MTA must run quickly and require less space. Moreover, it should be able to overcome denial of service (DoS) attacks. An ideal MTA must support the current mail transfer protocols; also it should be extensible to support possible future protocols. It accepts email from a local user or another MTA, and sends it either to local user or transfers it to another MTA. Also, it accepts email from an MTA to transfer to a third MTA. Therefore, an MTA has two accept inputs and two send outputs; as can be seen in Figure 3.3.

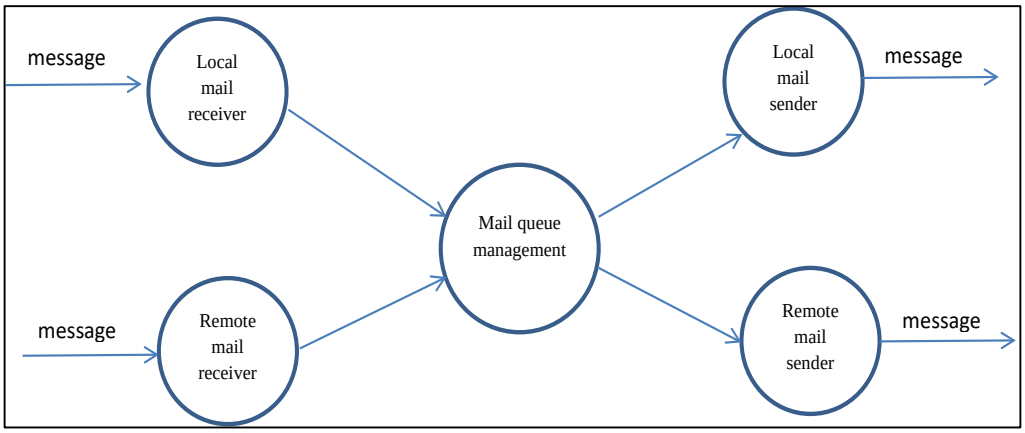


Figure 3.3 The functions of message transfer agent.

As clearly shown in Figure 3.3, an MTA consist of independent processes; but the sendmail performs all of them with a single software. Also, sendmail runs as root (super-user). If any failure occurs in any process of the system, then it leads to the exploitation of another processes of the system. Thus, separating each process from the rest of the MTA is a fine method to improve security. The qmail runs many processes with different qmail specific user identities. Only local deliveries process (qmail-lspawn) and starting process (qmail-start) run as root. The remote email receiver qmail-smtpd process executes with uid “qmaild”, the email queue management qmail-queue and qmail-clean execute with uid “qmailq” and qmail-send executes with uid “qmails”, the remote email sender qmail-rspawn and qmail-remote execute as qmailr. The qmail splits local email delivery into two processes, qmail-lspawn and qmail-local for security. Figure 3.4 shows the qmail structure and specific user identities.

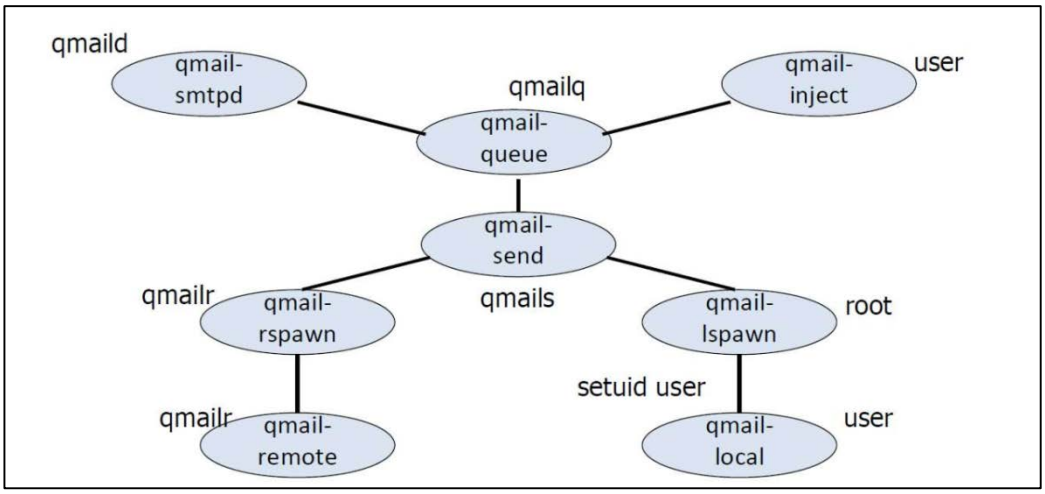


Figure 3.4 The structure of qmail and specific user identities.

The queue management is an important part of qmail. It must be reliable. The email is transferred in the email queue by qmail-queue. But, qmail-queue is not a process; it is a software which can always be called by many processes. It is triggered by the qmail-inject and by qmail-smtpd simultaneously. It gives each email a unique name to their files in the email queue directory to prevent any confusion about the messages. For this purpose, qmail uses the identity (ID) of the process which created the message, and then it changes the file name (message ID) to the i-node number of the filename as the file name. This solution prevents any confusion about email efficiently (Hafiz et al., 2004).

The commonly used mailbox format is ‘mailbox’. This type of format is used by sendmail and other MTAs. A single file is used for all received emails in this format. The main problem of the mailbox format is the excessive growth. Adding and deleting a message brings heavy load to the system. Also, two processes can deliver emails to the same mailbox at the same time, which may cause corruption. The solution is the ‘maildir’ format, which is used in qmail. This format is used by other MTAs like Exim, Postfix, Courier etc. A ‘maildir’ is a structured directory. This directory system can easily be used with a remote storage through Network File System (NFS) (RFC 7530, 2015). There are three subdirectories in the ‘maildir’, Their names are new, cur and tmp. The “new” directory includes newly delivered emails, which are still unread. The “cur” directory includes emails which have been fetched by the MUA program. The “tmp” is used for safely transfer from “new” to “cur” directory. The filename consists

of three parts separated by a dot: time, pid, host. The “time” is “unix time”. It is a count of the number of seconds passed since the beginning of the year 1970; pid denotes process ID of the software, and host denotes the host name of the server (Maildir, 1998). This format ensures that the file names are unique.

The qmail-local software offers a variety of facilities to the users. It can trigger a program or route to files as specified by the .qmail file in the home directory of the user. The .qmail file contains one or more lines. Each line has a delivery instruction. There are five types of delivery instructions: comment; program; forward; mbox; maildir. A comment line begins with a number “#” sign. It ignores this line. A program line begins with a vertical bar “|” (pipe character). It takes the rest of the line as a command to provide supply to the shell. A forward line begins with an ampersand “&”. It takes the rest of the line as an email address. A mbox line begins with a slash “/”, but does not end with a slash: ~./Mailbox. A maildir line begins with a slash “/”, and ends with a slash: ~./Maildir/. If .qmail has no lines, or does not exist, it follows the default delivery instructions set by the system administrator (qmail man page dot-qmail, 1998).

The modular structure of the qmail provides for the possibility of inserting a variety of programs to the qmail system. The qmail system can be seen in Figure 3.5. The SPF implementation can be executed in three different stages. The common implementation is at the SMTP stage by patching qmail-smtpd. But, there is a risk of losing emails due to a false positive decision. At this stage, the message has not been received. The second field is queue management. The qmail-queue program passes messages to the qmail-send. An SPF checking program is run by being inserted between the qmail-queue and qmail-send program. The third field is the local delivery stage (qmail-local). For this, dot-qmail configuration file is used. The first line of the .qmail file must be quarantine directory: ./Quarantine/. The second line must be the SPF checking program: |/spf_check.bin. The third line can be a Maildir directory. The SPF-passed messages are transferred here.

In summary, qmail is simpler, smaller and flexible software. The qmail has various

security layers. The structure of qmail is modular so that it decreases the risks of damage, caused by vulnerability. The module structure makes debugging procedure simple. It runs parallel processing efficiently. The codes of qmail eliminate buffer overflows. The qmail can be used in conjunction with vpopmail or LDAP in large-scale email companies.

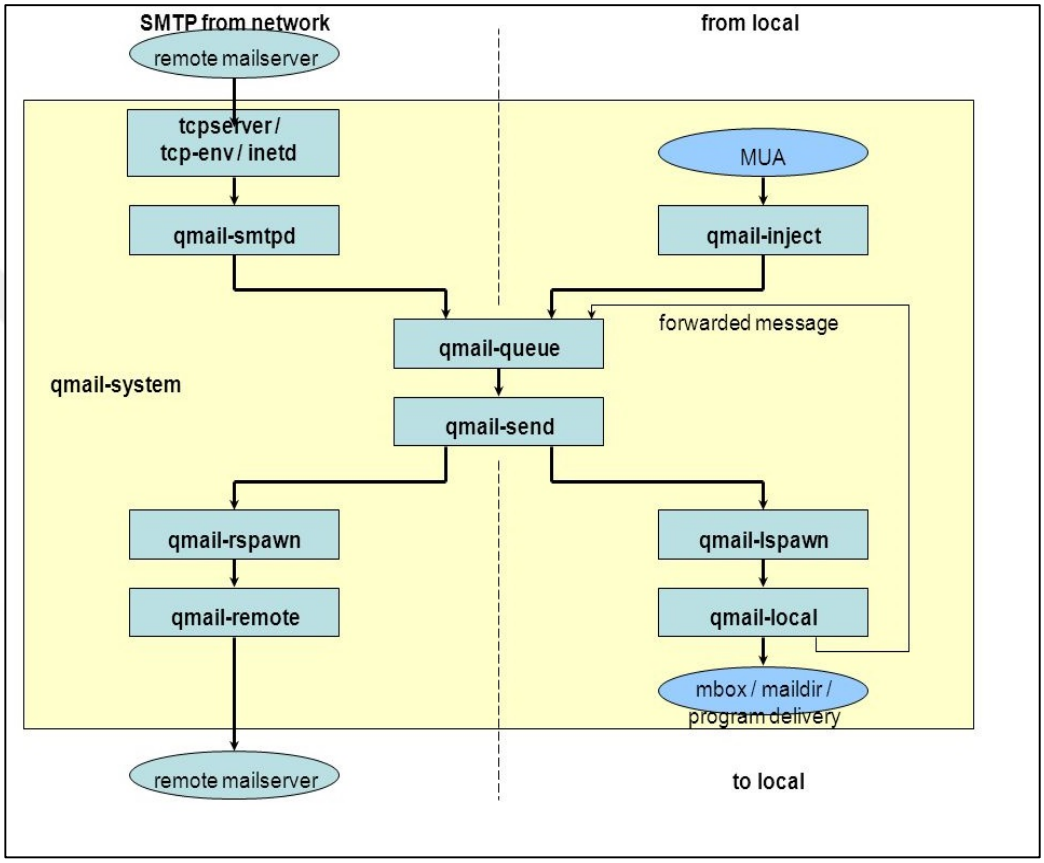


Figure 3.5 The big picture of qmail system (Oppermann, 1998).

3.5 Bayes' Theorem and Naïve Bayes Implementation

Bayes' theorem is a probability theory originally discovered by Thomas Bayes. However, Pierre-Simon Laplace developed and extended Bayes' theorem in 1774, unaware of Bayes' studies. It is a method to estimate future events by using the knowledge of past events (Koski & Noble, 2009). Bayes' theorem is used in various fields, such as medicine, sociology, economy and statistics. Many machine learning techniques are based on Bayes' theorem. Additionally, many specific techniques in statistics, such as calculating probability or interpreting medical results, are best

described in Bayes' theorem. In computer science, it is commonly used in text classification (Sahami et al., 1998). For instance, spam filters use Bayesian rules to determine whether an email is legitimate or spam, containing the words in the email.

Bayes' theorem provides a mathematical formula which uses the relationship between conditional probabilities. A conditional probability is a description of the possibility of an event given a fixed value for some other event.

Bayes' theorem is a direct application of conditional probabilities. The probability P(A|B), which means "A assuming B", is given by the formula:

$$P(A|B) = P(A \cap B) / P(B) \tag{3.1}$$

This is better written as:

$$P(A \cap B) = P(A|B) \cdot P(B) \tag{3.2}$$

On the other hand, P(B|A) of "B assuming A" is given by the formula:

$$P(B|A) = P(A \cap B) / P(A) \tag{3.3}$$

It is better written as:

$$P(A \cap B) = P(B|A) \cdot P(A) \tag{3.4}$$

The P(A∩B) on the left side is equal in the formula (3.2) and formula (3.4). It must be equal in right side, and we started with P(B|A) instead:

$$P(B|A) \cdot P(A) = P(A \cap B) = P(A|B) \cdot P(B). \tag{3.5}$$

Actually Bayes' theorem is:

$$\boldsymbol{P(A|B) = \frac{P(B|A)P(A)}{P(B)}} \tag{3.6}$$

P(A): The probability of the event A.

P(B): The probability of the event B.

P(A|B): An event A assuming that event B has occurred.

In practice, A denotes classes, and B denotes features. If number of feature increases (B_1, B_2), the formula will be;

$$\mathbf{P(A|}\boldsymbol{B_1}\mathbf{,}\boldsymbol{B_2}\mathbf{)} = \frac{\mathbf{P(}\boldsymbol{B_2}\mathbf{|}\boldsymbol{B_1}\mathbf{,A)P(}\boldsymbol{B_1}\mathbf{|A)}\boldsymbol{P(A)}}{\mathbf{P(}\boldsymbol{B_2}\mathbf{|}\boldsymbol{B_1}\mathbf{)}\boldsymbol{P(B_1)}} \tag{3.7}$$

If the number of properties increases, then the probability formula becomes more complicated.

3.5.1 Naïve Bayes (NB)

Bayes' theorem is simple only for two events. If the number of events increases, Bayes' formula becomes complicated. Naïve Bayes methods are a series of controlled learning algorithms, which are based on applying Bayes' theorem with the independent assumptions between each features. The presence of a particular feature is unrelated to the presence of any other feature. Namely it is accepted true, that the features are independent.

A Naïve Bayesian model is very simple for very large datasets, although its simplicity is widely used for complex classification methods.

Bayes' theorem ensures a method of computing the posterior probability, P(c|x), from P(c), P(x), and P(x|c). Naïve Bayes method supposes that the impact of the value of a predictor (x) on a given class (c) is independent of the values. This assumption is called conditional independence.

$$\mathbf{P(c|}\boldsymbol{x}\mathbf{)} = \frac{\mathbf{P(}\boldsymbol{x}\mathbf{|}\boldsymbol{c}\mathbf{)P(}\boldsymbol{c}\mathbf{)}}{\mathbf{P(}\boldsymbol{x}\mathbf{)}} \tag{3.8}$$

If there are n predictors (features)

$$P(c|X) = P(x_1|c) \cdot P(x_2|c) \cdot \ldots \cdot P(x_n|c) \cdot P(c) \quad (3.9)$$

In the formula above:

P(c|x) is the latter (posterior) probability of class given predictor (feature).

P(c) is the previous (prior) probability of class c.

P(x|c) is the (possibility) likelihood that is the probability of predictor given class.

P(x) is the previous probability of predictor.

As a result of this calculation, the class is selected with the largest posterior probability. This is known as the maximum a posteriori (MAP) decision rule.

Another form of formula is:

$$\boldsymbol{P(C|f_1, f_2, \ldots\ldots, f_n) = \prod_{i=1}^{n} P(fi|C)} \quad (3.10)$$

In the formula (3.10), C denotes class, and f denotes feature. Terms of each product are features. In spam classification methods, the frequency of words is calculated.

CHAPTER FOUR
SPAM FILTERING USING SENDER AUTHENTICATION NETWORK

4.1 Overview

As mentioned in the previous section, there are two deficiencies of the SPF implementation. Firstly, the SPF usage ratio is insufficient (27.3%); and every policy except “-all” is not very useful. Only the policy "-all" is a necessary condition for rejecting the message from the recipient mail server. However, the ratio of “-all” is low. The stats on the site “spf-all.com” indicate 31.3%, and this is 8.55% of over 88 million domains. To solve this problem, we proposed an estimation method called “Sender Authentication Network”.

The second deficiency is the rising of malicious usage. For the second problem, we have developed a Naïve Bayes implementation over the SPF terms. The proposed method will be described in the next section.

In this section, the solution to the first problem mentioned above is explained. The implementation of this solution is explained in the next sections.

4.2 Sender Authentication Network

The Sender Authentication Network (SAN) is a method which predicts SPF record of domains having no SPF record. This method is similar to the best-guess SPF record (SPF best-guess, 2010). Best-guess is an old (2005) recommendation of SPF developers, and it works for guessing the IP address range of a domain's outgoing mail servers. If a domain has no SPF record, its SPF record is guessed as the following:

```
“v=spf1 a/24 mx/24 ptr ?all”
```

In the best-guess method, it is assumed that the sending IP addresses of legitimate servers are in the same C-class network with the host and MX’es of related domain. The “ptr” is used for the reverse DNS lookup in this domain. The qualifier "?" means

a "Neutral" result, which means there is no policy. Since guessing an SPF record can decrease the confidence of SPF test results, it is revoked by SPF community, and the practice is not recommended.

But, there are two differences between our method and SPF best-guess. First is the prediction size. The prediction size is important and impact the result; a narrower prediction increases FP while decreases FN, and a wide range of prediction decreases FP, while increases FN. Secondly, the prefix of “all” is “?” (? means a "Neutral" result), and in our method the prefix of “all” is “-” (-all: reject).

4.3 The Size of the Sender Authentication Network

Many legitimate emails have a relationship between the sender IP and the sender domains, since the distribution of the IP addresses follows a hierarchical scheme. In this hierarchy, the Internet Corporation for Assigned Names and Numbers (ICANN) is responsible for managing the IP address spaces. First octet is the Regional Internet Registry (American Registry for Internet Numbers (ARIN), European IP Networks (RIPE), Asia Pacific Network Information Centre (APNIC), Latin American and Caribbean Network Information Centre (LACNIC), African Network Information Centre (AFRINIC), then Local Internet Registry (Internet Service Provider, University, large firms, Telco), and finally end user. Thus, the sending IP address and the IP addresses of the “A” and “MX” records are usually in the same network range. We use this relationship to predict SPF records.

The term “network” in the title “Sender Authentication Network” refers to the network size. The size in the SPF best-guess record is C class (/24). To decide the optimal size, some comparisons were made between SPF records and “A”, ”MX” records in the DNS system.

A number of abbreviations has been made in the thesis; IP2 indicates first two parts of the IPv4 addresses and IP3 indicates first three parts of the IPv4 addresses. So, IP2

indicates /16 netblock, and IP3 indicates /24 netblock in the Classless Inter-Domain Routing (CIDR) notation (RFC 4632, 2006).

4.3.1 The Comparison Method

To evaluate the correct prediction rate, we used lots of domain names having an SPF records. These domain names have been collected from the Internet and email server logs. A lot of DNS queries have been performed for every domain name, to find IP addresses of the A, MX and SPF (TXT) records. A database was used for comparison. Two different comparisons were made: IP2 and IP3. For IP2, first two octets and for IP3, first three octets of the IP addresses were used. For each comparison, the values (IP2 and IP3) were inserted to the table. Comparison process is carried out as follows: For the IP2 comparison, the domains, which have SPF record containing greater coverage than B class networks (IP2), have been discarded. Big companies may be a suitable example for this situation. The domain “apple.com” has an SPF record: that is “v=spf1 ip4:17.0.0.0/8”. Their ratio is smaller than 0.1%, so it does not affect the overall rate. For a domain name, if every record (IP address group) in the SPF columns is located in any A/MX columns, we can say that A/MX records cover every SPF records completely. This result is defined “full coverage”. But, some records in SPF columns can be in A/MX columns and some not. Therefore, A/MX records contain SPF records partially. This result is defined “partial coverage”. If any SPF record exists in A/MX columns, this situation is defined “null coverage”.

In the following example, the IP addresses in the A/MX records cover the IP addresses of the SPF records completely both for IP3 and IP2 comparisons. For IP3, SPF has the addresses 11.22.33 and 11.22.44, and these are in MX and A records also. For IP2, SPF has the address 11.22 and it covers IP addresses in the MX and A records.

- `MX records: 11.22.32.14 11.22.33.15 11.22.33.16`
- `A records: 11.22.44.17 11.22.44.20`
- `SPF records: v=spf1 ip4:11.22.44.0/24 ip4:11.22.33.0/24 -all`
- `IP3 network: v=spf1 ip4:11.22.33.0/24 ip4:11.22.44.0/24 -all`
- `IP2 network: v=spf1 ip4:11.22.0.0/16 –all`

In another example, the IP addresses in the A/MX records cover the IP addresses of the SPF records completely for IP2 comparisons. Also for IP3, there is a partial coverage.

- `MX records: 11.22.33.14 11.22.33.15 11.22.33.16`
- `A records: 11.22.33.17 11.22.33.20`
- `SPF records: v=spf1 ip4:11.22.44.0/24 ip4:11.22.33.0/24 -all`
- `IP3 network: v=spf1 ip4:11.22.33.0/24 ip4:11.22.44.0/24 -all`
- `IP2 network: v=spf1 ip4:11.22.0.0/16 –all`

In the example above, the A/MX records (1.2.3.0/24) do not contain the SPF record (1.2.4.0/24) for IP3 comparison. But, the A/MX records (1.2.0.0/16) contain the SPF records (1.2.3.0/24 and 1.2.4.0/24).

Three comparisons were made prior to 2009 March (Dalkilic et al., 2009). We had used the domain names of the email, which were sent to the deu.edu.tr domain. These domains had SPF records, of course. The results of the IP2 comparison are shown in Table 4.1.

Table 4.1 The comparison result prior to 2009 March.

Date of Comparison	Domain Number	Full Coverage Ratio(%)	Partial Coverage Ratio(%)	Null Coverage Ratio(%)
Jan 2007	2,074	51.70	41.00	7.30
May 2008	206,804	64.11	31.76	4.13
Jan 2009	253,347	66.34	30.60	3.06

The data collection and comparison was done in January of 2007 for the first time. 2074 domain names were processed; the full coverage ratio was 51.70% and null-coverage was 7.30%. This result encouraged us to continue with our method and work on improvements. The second analysis was made in May of 2008 and the third was in January of 2009. The result of May 2008 shows that 64% of SPF records (IP addresses) were fully covered by the IP networks of A/MX records. In January 2009, this number increased to 66%. The increase in this ratio, parallel to the increase of domain number, is promising.

The next comparison was made in November 2009. In this analysis, two datasets (domain names) were used. One of them was collected from deu.edu.tr email server, and the other was collected from spf-all.com. 35,830 domains were in both datasets. In the previous analysis, only IP2 comparison was made. But, in the following analysis extra IP3 comparison was added.

4.3.1.1 The Dataset from deu.edu.tr Email Server

In this dataset, 674,932 domains having SPF record have been collected from January 2007 to November 2009. However, at that time (November 2009) during the execution of comparison, 183,625 domains had no SPF records, and 35,310 domains (7.1%) had empty SPF record (v=spf1 -all). It means these domains had never sent email. The remaining 455,997 domains were compared. 379 domains had SPF record with greater than IP2 network. Example: apple.com "v=spf1 ip4:17.0.0.0/8 ~all". 144,998 domains had SPF record with greater IP3 network. After IP2 comparison, 40.2% of 455,997 domains' predicted SPF records covered SPF records completely. Partial coverage ratio was 45.2%, and null coverage ratio was 14.6%. After IP2 comparison 65.9% of 455,997 domains' predicted SPF records covered SPF records completely. Partial coverage ratio was 32.4%, and null coverage ratio was 1.7%. The comparison results are shown in Table 4.2.

Table 4.2 The comparison results.

Date	Dataset	Domain number	C. T.	Full Cov. (%)	Partial Cov. (%)	Null Cov. (%)
Nov 2009	deu.edu.tr	459,165	IP2	65.90	32.42	1.68
Nov 2009	deu.edu.tr	459,165	IP3	40.20	45.20	14.60
Nov 2009	spf-all	236,513	IP2	70.10	26.22	3.66
Nov 2009	spf-all	236,513	IP3	64.30	32.10	3.60

4.3.1.2 The Dataset from spf-all.com

In November 2009, the number of domains was 1,155,984 (with "-all") in the list of the spf-all.com. But approximately half of them (578,441) did not have an SPF record. 577,543 domains had SPF record and 341,023 domains (59%) had empty SPF (v=spf1 –all). Remaining 236,513 domains were compared. 195 domains had SPF record with greater IP2 network. 29,733 domains had SPF record with greater IP3 network. After IP3 comparison, 64.3% of 236,513 domains' predicted SPF records covered SPF records completely. Partially coverage ratio was 32.1%, and null coverage ratio was 3.6%. After IP2 comparison 70.7% of 455,997 domains' predicted SPF records cover SPF records completely. Partial coverage ratio was 27.8%, and null coverage ratio was 1.5%. The comparison results are shown in Table 4.2.

The fourth column in Table 4.2 denotes the comparison type IP2 or IP3. The fifth column denotes the ratio of the domains where the addresses in A and MX records fully cover the IP2 and IP3 network blocks indicated in the SPF records. In the following columns, the ratios for the partial and null coverage are listed.

4.3.2 The Results of Comparison

Based on these tests, IP2 comparison is better for FP reduction. These results are summarized in the Table 4.2. The correct prediction rate of this method is nearly between 65% (only full coverage) and 96% (partial coverage + full coverage). We can categorize the domain names, based on the network size in their SPF record. The full coverage ratios are found per each network size. The result of this work is shown in Table 4.3.

First column in Table 4.3 indicates the number of network blocks defined in the SPF records. When SPF records of 2,774,535 domains are considered, 59.92% of them are only one IP2 network block. A/MX records of those domains fully cover the specified IP2 network block with a ratio of 99.11%. In the IP3 comparison, 55.21% of them are only one IP3 network block, and full coverage ratios are 93.93%. The ratios

are shown in Table 4.3. FCR means full coverage ratio. The SPF records having 5 or less unique IP2 network blocks form 88.52% of all the records, and for the IP3 network blocks, this number is 83.09%. If the count of the network blocks increases, full coverage ratio decreases. The remaining domains have 6 or more unique network blocks defined and the full coverage ratio of the A/MX records with these records decreases nearly to zero. Another result in this table is that, the FCR is better for smaller companies, because they have fewer IP addresses in the same C class network.

Table 4.3 The detailed comparison results

Netblock number	**IP2 netblock %**	**FCR of IP2**	**IP3 netblock %**	**FCR of IP3**
1	59.92	99.11	55.21	93.93
2	15.85	45.37	13.72	58.22
3	5.37	23.03	5.89	17.93
4	3.03	2.17	3.46	12.85
5	4.35	0.17	4.81	2.75

4.4 The Description of Sender Authentication Network

The sender authentication network is an IP2 (B-class) address range, which covers the resolved IP addresses in the DNS records A, MX and SPF of sender domain name. This network includes each different IP2 networks. For example; the domain name “odtu.edu.tr” did not have an SPF record up until this thesis was being written. Its “MX” records are “144.122.144.157”, “144.122.144.163” and “A” record is “144.122.144.137”. The “Sender Authentication Network” of “odtu.edu.tr” is 144.122.0.0/16. Then we create virtual SPF record for this domain as `“v=spf1 ip4:144.122.0.0/16 -all”`. If the domain name has an SPF record, then it is formed in the following example: In this example, “itu.edu.tr” has SPF record. After DNS query we found its MX records are “160.75.25.103” and “160.75.25.104”, its A records are “160.75.175.209” and “160.75.175.208”, and its SPF record is `“v=spf1 ip4:160.75.25.118 ip4:160.75.25.119 -all”`. Then we create the “Sender

Authentication Network" of the domain "itu.edu.tr" is 160.75.0.0/16. As a result, we expand the SPF record of the "itu.edu.tr".

In some cases, domain names may be insufficient and SAN may not cover all legitimate IP addresses. This situation can cause FP. Therefore, to reduce the number of FPs, we have to find related domain names with the sender domain. In order to solve this problem, new domain names are found, which are associated with the sender domain. These are parent domains and subdomains. The sender domain name can either be a parent or a subdomain. For a parent domain, it is extended to commonly used subdomain prefixes like "outgoing, mail, smtp, mx" etc. If the domain is a subdomain, then we use the upper domain name. We have a problem when deciding whether it is a subdomain or a parent domain. For the generic TLD's (com, net, org, etc.), the parent domain name is second level (domain.com). Each country has different distribution policies. Some countries use second level: domain.de, domain.fr. Some others use third level: domain.co.uk, domain.com.tr. upper domain names (com, net, com.tr, co.uk) have not DNS records (A, MX, SPF). Therefore, domain names can be multiplied in two directions. If the domain is a.b.c, then we need to look at the subdomain names that are: "mail.a.b.c, mx.a.b.c, smtp.a.b.c", and the parent domain is "b.c". This is explained in the following example:

Sender domain = engineering.mit.edu
Sender IP2= 18.72
DNS (A, MX) IP2= 18.9
Upper domain: mit.edu
SAN of upper domain IP2= 18.72, 18.7, 18.9.

If we apply SAN over the domain "engineering.mit.edu", our method says that the incoming email from the IP address "18.72.x.x" is spam. If we use the expanded parent domain name "mit.edu", our method says that the incoming email from the IP address "18.72.x.x" is legitimate.

CHAPTER FIVE
IMPLEMENTATION OF THE SENDER AUTHENTICATION NETWORK

5.1 Overview

This chapter explains the implementation of the spam filtering system. First the system architecture of the server, operating system, MTA server (qmail) and database server are informed. Secondly, the details of the filtering system and qmail are explained. Thirdly, the results of the implementation such as spam statistics; false positives tables, user surveys are detailed. Fourthly, SpamAssassin and this method are compared. Finally, the weaknesses of the method are explained.

5.2 The Structure and Operation of the Method

We began the SAN project in 2007. We used the Solaris 10 operating system, MySQL as the database server, and qmail as the MTA server. Unlike traditional MTAs (like sendmail) qmail has a modular structure as we mentioned in Section 3.4. Email service is divided into functions. Each function is separate software. We implemented SAN in qmail system at local delivery stage through ".qmail" file. Every incoming email is accepted by qmail-smtpd. In the next stage, the email is passed to the mail queue, and lastly, email is delivered to the user's mailbox. Our software runs before the last stage through the ".qmail" file. The flowchart of qmail is shown in Fig. 3.5. The contents of the file ".qmail" are as follows:

```
./Quarantine/  "./" means that emails are copied to the directory.
|/path/of/first_filter_SAN   "|" means a software is implemented over the email.
|/path/of/second_filter (Optional)
./Maildir/
```

The operation of the spam filtering system is as follows:

1. Each incoming email is placed in the user's quarantine zone at first.
2. Header check is carried out from the software on the first two octets.

3. Ham (non-spam) emails are passed to user inboxes.

4. The sender address, date, subject and the IP2 values of the spam email are written in a file and sent to the corresponding user as a report (There is already a copy in the quarantine).

5. The user sends a request email to a predefined mailbox if an email is doubted to be a ham email.

6. Upon receiving the request email, small software (shell script) recovers the original copy of the email from the quarantine and transfers it to the user's mailbox.

7. After reading this email, if the user decides that this email is actually not spam, they specifiy this mail as a false-positive by making a feedback.

8. The domain and the IP2 of the reported false-positive email are appended to the whitelist.

9. The emails are kept in the quarantine for three months, and at the end of this period they are erased (Dalkilic et al., 2009).

Each incoming email is accepted by MX server; and it is passed to the main email server. This step can be seen in the line beginning with "Received:" of the email header. Example: "Received: from unknown (HELO mail-wm0-f51.google.com) (74.125.82.51) by gelen.posta.deu.edu.tr with SMTP; 2 Aug 2016 12:36:19 -0000". In this example, the host "gelen.posta.deu.edu.tr" is an MX server; and 74.125.82.51 is the IP address of the sender. The domain name is retrieved from the sender address, and IP2 is extracted from the IP address.

The structure of the software is simple. The flowchart of this software is shown in Figure 5.1. The database (mysql) that we use in the software contains columns for IP2 and domain information. For preparation, we create an empty whitelist and blacklist table. In the first step, the software retrieves the sender domain name and the IP2 of the sending IP address. The sending IP address is retrieved from "Received:" field of the email header, which is communicating with the MX server (gelen.posta.deu.edu.tr).

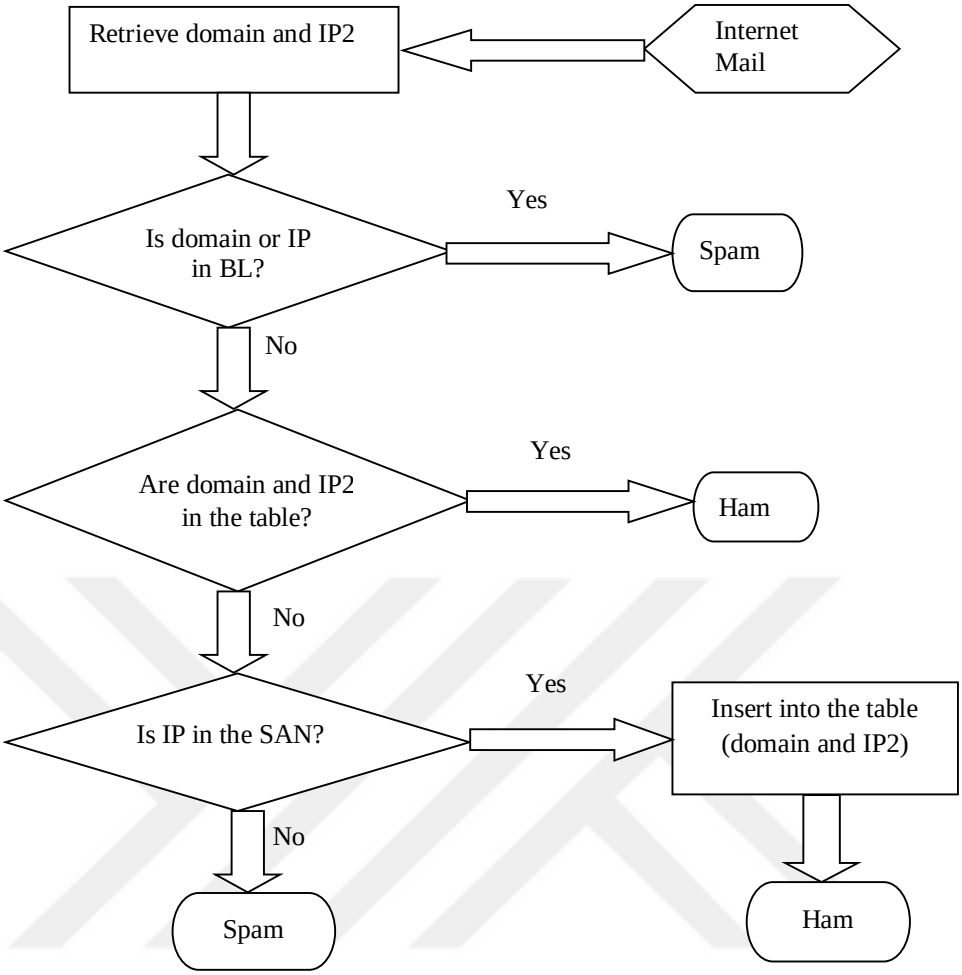


Figure 5.1 The flowchart of the program.

In the second stage, the software analyzes the sender address, domain name and sender IP address. It checks whether one of them is in the blacklist table or not. If at least one of them is in the blacklist table, this email is a spam. Then, the domain name and IP2 address are queried from the whitelist table to decrease DNS queries. In the last step, it creates SAN. It resolves with DNS query SPF, “A” and “MX” records. It compares the sender IP2 with the SAN’s IP2’s. If at least one of the IP2 addresses of the SAN is same with the IP2 of the sender, then the email is ham (non-spam); otherwise, it is spam. If it is ham, then the sender IP2 and domain name are added to the whitelist table. FP feedbacks, which are coming from user, are also inserted to the table.

Until 2009, the SAN did not contain the domains having SPF records, and the software used to check the SPF before the SAN test. This this study was published in the SIN 2009 conference (Dalkılıç et al., 2009). There were two reasons for the modification. According to the users' feedbacks, some ham emails had been incorrectly classified as spam (false-positive). Even though sender IP addresses were in the SAN of the sender domain, these emails were detected as spam by the SPF policy. Secondly, SPF tests did nothing for many emails, whose SPF records contain "+all", "~all", and "?all"; namely the absence of the term "-all". The only policy that works is the "-all" (fail) policy. Otherwise the SPF protocol does not generate any decisions.

5.3 The Results of the Implementation

The spam filtering software saves the date, IP address, sender address and SPF status for each email into a file. Thus, we can create detailed statistics through this file. Additionally, Symantec spam reports were used for comparison.

5.3.1 The Statistics of the Results

Spam ratios of our study (DEU) and Symantec's global spam ratios are shown in Table 5.1 for the last nine years (2007-2015). In this table, the number of ham and spam emails, the number of users and the rates of spam are annually given. Also, Symantec spam ratios were added to this table for comparison. Local emails (from deu.edu.tr to deu.edu.tr) were excluded.

The number of users (volunteers) was 44 in January 2007, 2,354 in January 2008, 3,544 in December 2008, 3,838 in December 2009, 3,989 in December 2010, 4114 in December 2013, 4186 in December 2014 and 4223 in December 2015. This number is 70% of active email users in deu.edu.tr domain. Active email users mean who check their mailbox at least once a week. Total number of spam e-mails is 13,845,137 and 21,056,013 for the years 2007 and 2013 respectively. In 2013, our spam ratio was 63.03% compared to Symantec's 66.43%.

Table 5.1 The spam statistics.

Date	Number of Users	Number of Ham emails	Number of Spam emails	Spam Ratio from DEU	Spam Ratio from Symantec
2007	2,184	1,670,926	13,845,137	89.23	68.46
2008	3,544	4,276,611	57,004,566	93.02	77.45
2009	3,838	5,786,053	45,831,178	88.79	86.91
2010	3,989	7,555,774	52,916,948	87.51	88.78
2011	4,022	8,860,670	41,628,834	82.45	74.94
2012	4,058	11,218,201	27,452,376	70.99	68.42
2013	4,114	12,348,904	21,056,013	63.03	66.43
2014	4,186	13,270,357	10,955,458	45.22	66.03
2015	4,223	11,783,975	8,518,176	41.95	52.73

The average spam rates (spam / (spam+ham)) were 89.23% in 2007, 93.02% in 2008, 88.79% in 2009, and 60.03% in 2013. After 2010, these rates began to decrease. The effect of global politics of states is behind this. For example, the ISPs in Turkey block the TCP port 25 for home users (Turk Telekom, 2009).

The spams coming to our email server are showing parallelism with the global rates. After improving our method in 2009, our spam rates' downturns have become compatible with the results of Symantec. From 2010 to 2015, spam ratios decreased in both our results and Symantec's. The comparative graphics are shown in Figure 5.2.

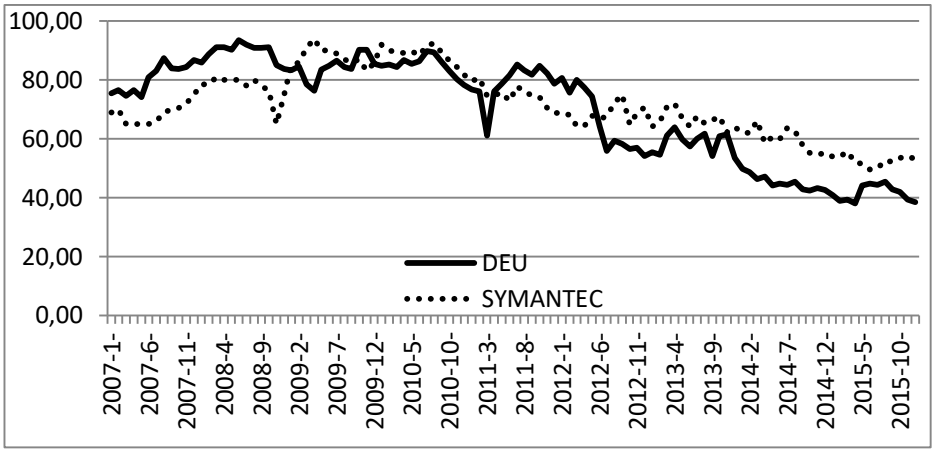


Figure 5.2 The spam statistic from DEU and Symantec.

Using the software logs, we can find SPF utilization ratios in ham and spam emails. These statistics are shown in Table 5.2. In 2007, the SPF usage ratio of ham emails is 45.58% and for spam emails 25.38%. In 2013, the SPF usage ratio of ham emails is 78.67% and for spam emails 43.92%. In 2013, the ratio of SPF utilization in email server domains is greater than the previous 6 years. The SPF ratios in ham emails are greater than the spam emails for 7 years. SPF ratio in ham email was greater (nearly twice) than SPF ratio in spam mails. After analyzing the emails, it is seen that, the users receive most of their ham emails from the giant email service providers such as hotmail.com, gmail.com, and yahoo.com, which have SPF records.

Table 5.2 The SPF ratios per ham and spam emails.

Year	**Ham emails' SPF ratio (%)**	**Spam emails' SPF ratio (%)**
2007	45.58	25.38
2008	46.07	15.79
2009	53.28	27.07
2010	65.62	25.72
2011	71.33	26.23
2012	44.28	26.46
2013	78.67	43.92
2014	75.45	53.24
2015	79.12	66.56

The spam and ham distribution based on the day of the week (the data is the average of 8 weeks randomly chosen from the year 2009) are shown in Figure 5.3. Ham ratios decrease on the weekends significantly. The average ratio in working days is 18.79%, on Saturdays it is 9.19%, and on Sundays it is 6.88%. But, spam rates are on average in the range of 13.00% – 15.50%.

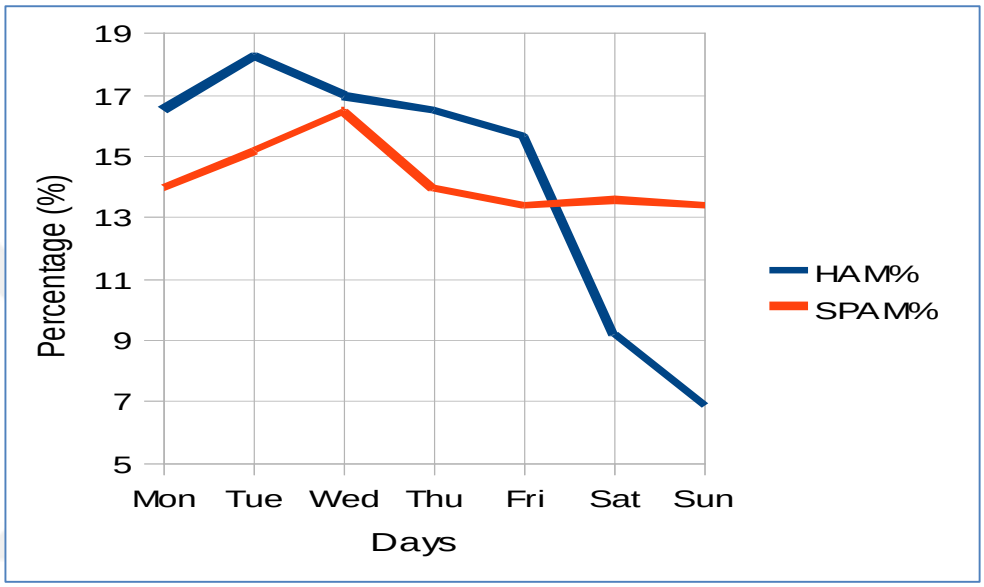


Figure 5.3 The spam rates distribution per days of the week.

5.3.2 The False-Positive Feedbacks and Whitelist

If a user doubts a ham email is marked as spam email, he/she sends a request email to a predefined mailbox (postmaster of quarantine). Upon receiving the request email, small software (shell script) recovers the original copy of the email from the quarantine and transfers it to the user's mailbox. After reading this email, if the user decides that this email is actually not spam, then the user marks this email as false-positive by making a feedback. These processes are logged in a file. In Table 5.3, the false-positive (FP) suspicious email count (FP-doubt), the actual FP email count (FP emails), FP ratio, and row count of whitelist are indicated. If one user tries to see a spam email, then the software saves this action. We defined this action as FP-doubt. The count of FP-doubt decreased dramatically in 2009. The number is 21,269 in 2008, and 3660 in

2009. This is because, the “Subject” of spam emails is added to the spam information, which was regularly sent to the user.

The total number of user who wanted to see at least one spam-marked email was 1348 (30% of the total email users) by the end of 2015. Also, 70% of total users do not have FP feedbacks. When the same action is repeated more than one, it means, this email (spam-marked) is actually ham.

Table 5.3 The false-positive information.

Date	FP-Doubt	FP Emails	FP Ratio %	Whitelist Entries
2007	16,282	1204	0.0721	219
2008	21,269	878	0.0205	219
2009	3660	1505	0.0260	234
2010	3721	1712	0.0227	265
2011	3535	1214	0.0137	261
2012	4439	1637	0.0145	284
2013	4583	1621	0.0131	276
2014	4968	1551	0.0117	149
2015	3928	1283	0.0109	185

The false-positive ratio (Pantel and Lin, 1998) is calculated in Eq. (5.1)

$$R_{fp} = 1 - \frac{CoreCount(NonSpam)}{TrueCount(NonSpam)} \tag{5.1}$$

Where R_{fp} indicates false-positive ratio; CorCount(NonSpam) means number of email which are classified as non-spam(ham) by the software; TrueCount(NonSpam) is the number of FP emails + CorCount(NonSpam).

The numerator in Eq. (5.1) is the ham classified email count, and the denominator is the total of the FP email and non-spam classified email counts. Eq. (5.1) can be simplified as Eq. (5.2):

$$R_{fp} = \frac{FalsePositiveCount}{TrueCount(NonSpam)} \quad (5.2)$$

FP-rates are calculated by dividing the number of FP emails (Table 5.3 Row 3 and 7, Col 3) by the number of total ham emails (Table 5.1 Row 3 and 7, Col 3). For 2009 and 2013, these values are calculated as follows:

2009: 1505 / 5,786,053 = 0.00026.

2013: 1621 / 12,384,904 = 0.00013.

5.3.3 The Questionnaire Results

Three questionnaires were carried out among the volunteers. First two were conducted in December 2007 with 510 people and in June 2008 with 684 people, and the results were mentioned in (Dalkilic et al., 2009). The third questionnaire was done in November 2009 with 648 people. Only 25% of the participants work in administrative units, the others (75%) are academics. 81% of them were using the software for more than one year. They answered daily FN number, and monthly FP number. According to users' feedback, the lowest unit of time for FN is day and for FP, it is month.

First questionnaire was carried out in December of 2007 among 510 people. Only one quarter of the volunteers participated in the first questionnaire. The second questionnaire took place in June 2008 and 684 volunteers, 20% users of the total by then, participated. The same questions were asked. The first question asked to the volunteers was whether they were happy about volunteering the program. It was answered positive by all users. The second question was the success rate of blocking spam. At the first questionnaire, the feed-back ratio came out to be 97.35% on the average. This value was 97.41% at the second questionnaire. The results of the first and second questionnaires are given in Table 5.4. Approximately half of the participants (44%, 49%) said that the success rate of blocking spam was excellent.

Table 5.4 Blocking ratios declarations.

Voted Success Rate (%)	Ratio of Participants (%)	
	Dec. 2007 Period	June 2008 Period
100	44	49
99	16	14
98	12	11
97	2	3
96	2	1
95	13	12
94	1	0
90	7	6
85	0	1
80	2	1
75	0	1
70	1	1

The third question was about the average false-positive declarations. The number of false-positive emails was reported as 2.17/month on the average at the first questionnaire. This value was 1.74/month at the second questionnaire. This value is a number and not a percentage. The results are shown in Table 5.5.

Table 5.5 Number of false-positive declarations.

Monthly Av. Number of False-positive Declarations	Ratio of Participants Dec. 2007 Period	Ratio of Participants June 2008 Period (%)
0	55	57
1	13	15
2	9	9
3	6	5
4	3	2
5	5	5
6	1	1
7	1	1
8	1	0
9	0	0
10	3	3
20	1	1
30	2	1

Approximately more than half of the participants (55%, 57%) said that the success rate of blocking spam was excellent.

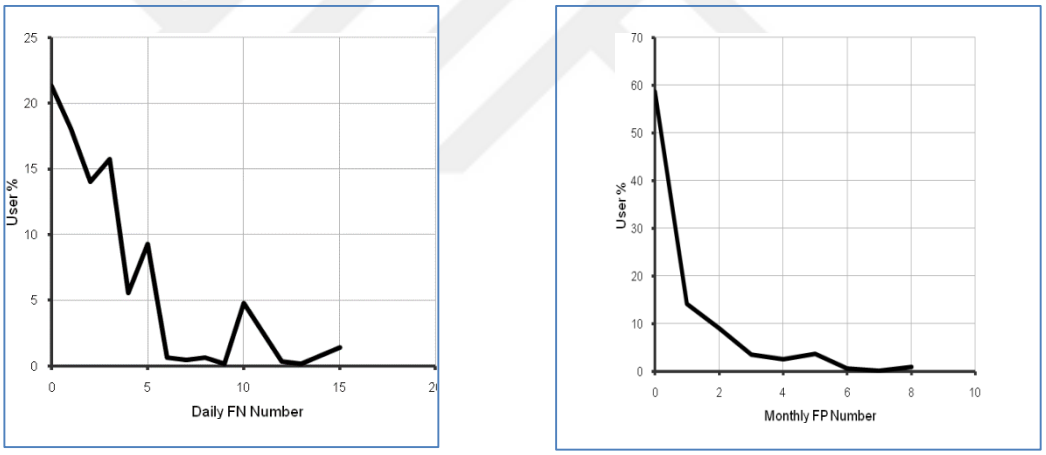
The third questionnaire was done in November 2009 with 648 people. Only 25% of the participants work in administrative units, the others (75%) are academics. 81% of them were using the software for more than one year. They answered daily FN number, and monthly FP number. According to users' feedback, the lowest unit of time for the FN is "day" and for the FP it is "month". The results are displayed in Table 5.6.

Table 5.6 Number of false-positive declarations.

Daily FN Number	**Number of User**	**Percentage %**	**Monthly FP Number**	**Number of User**	**Percentage %**
0	138	21.30	0	380	58.64
1	117	18.06	1	92	14.20
2	91	14.04	2	58	8.95
3	102	15.74	3	23	3.55
4	36	5.56	4	16	2.47
5	60	9.26	5	24	3.70
6	4	0.62	6	4	0.62
7	3	0.46	7	1	0.15
8	4	0.62	8	6	0.93
9	1	0.15	9	2	0.31
10	31	4.78	10	14	2.16
12	2	0.31	12	2	0.31
13	1	0.15	13	1	0.15
15	9	1.39	15	0	0.00
18	2	0.31	18	0	0.00
20	14	2.16	20	3	0.46
22	1	0.15	22	0	0.00
23	1	0.15	23	1	0.15
25	9	1.39	25	2	0.31
26	1	0.15	26	0	0.00
28	1	0.15	28	0	0.00
29	2	0.31	29	4	0.62
30	18	2.78	30	15	2.31
Total	648	100.00	**Total**	648	100.00

The median of daily FN is 2 (21+18+14=53). First quartile is 1 (25%), and the third quartile is 5 (75%). The interquartile range is 5-1=4. According to Tukey's outlier method (Tukey, 1977), the numbers, which are greater than 5+3*4=17, are outliers. The rows of 18 and more are discarded.

The median of monthly FP is 0. First quartile is 0 (25%), and third quartile is 2 (75%). The interquartile range is 2-0=2. According to Tukey's outlier method, the numbers, which are greater than 2+3*2=8, are outliers. The rows of 9 and more are discarded. Figure 5.4 shows the questionnaire results. Figure 5.4(a) shows the graphics of daily FN number and Figure 5.4(b) shows monthly FP number. The y axis represents the percentage of users. The x axis represents daily FN number in Figure 5.4, monthly FP number in Figure 5.5. The results of these graphics show that participants were satisfied.



(a) (b)

Figure 5.4 The questionnaire results (a) daily FN number; (b) monthly FP number.

After removal of outliers, FP rate was calculated as 0.6%, and FN rate was calculated as 4.2%. This calculation was done with numbers in November 2009. This FP rate is different from FP rate at the Table 5.3. The reason is that, very satisfied volunteers (zero FP and little FN) have not participated to the questionnaire. Only one quarter (970) of the total volunteers (3838) had a FP doubt beginning from 2007 to the end of 2009. But, the ratio among those who have participated in the questionnaire is 50%.

5.4 Comparison with SpamAssassin

The efficiency of a method can only be measured by comparing with other methods. The method to be used for comparison must be easily accessible by others. SpamAssassin (SA) is popular and commonly used open source anti-spam software (SpamAssassin, 2004). Therefore, SA is used for comparison purposes with several spam filtering techniques (Cook et al., 2008; Gargiulo and Sansone, 2008). Our aim is to compare SA with SAN and collaboration of the two softwares: First filter is SAN; and the second filter is SA.

SA uses many tests to determine whether an email is spam or not. Each test has a score, which is a positive or a negative value. While the positive values increase the probability of being spam, negative values decrease the probability of being spam. If the sum of these values is higher than the threshold (required_score), then the message is considered as spam.

We made two comparisons with an interval of 3 years. First comparison was made in 2011, and second was started in 2014 and ended 2015. In the first comparison we used 30 email addresses. We selected these addresses from those, who make frequently feedbacks as FN and FP. Incoming emails were analyzed for 6 months (April-September) in 2011. The computer system was Pentium 4 machine (3.0 GHz), with 1 GB memory. We used FreeBSD 6.3 as an operating system, qmail as MTA server, and SA version was 3.2.5.

In this comparison 83,494 ham and 306,413 spam emails have been tested by SA and SAN. In the SA test 2 parallel operations were performed because of CPU load. In the SAN test 10 parallel operations were performed at the same time, and without using any whitelist. During this SA test default configuration were used. The results of comparison are given in Table 5.7.

The average execution time of SA was 5.76 sec for ham email, and 5.18 sec for spam email. The average execution time of SAN was 0.82 sec for ham email, and 1.8 sec for spam email. The load average of CPU was 0.70 for SA, and 0.30 for SAN. The

FP rate of the SA test was 10.79% averagely, and the FN rate was 22.39%. In the SAN test the FP rate was 4.25% averagely, and FN rate was 1.44%. If we used whitelist, the FP rate of SAN was 0.18%.

Then we tested interoperability of the two methods. We used SAN as first filter. SA was applied to the mail that passed the first filter SAN. Spam filtering with series filters decreases FN, but it increases FP. To minimize FP emails, we changed the required_score value to 12. After this modification, except 5 users this FP value did not change. The FN ratio was decreased from 1.44% to 0.49%; it decreased by 66%.

Table 5.7 Summary of the first comparison of SAN and SpamAssassin.

Comparison criteria	**Our Program**	**SA**
HAM email number	83,494	83,494
SPAM email number	306,413	306,413
Number of concurrent executed addresses	10	2
CPU load average	0.3	0.7
FP ratio %	4.25	10.79
FN ratio %	1.44	22.39
time for HAM (second)	0.82	5.76
time for SPAM (second)	1.8	5.18

In the second comparison, 36 email addresses were used. These addresses belong to users who frequently give feedback FP and FN. Incoming emails to those addresses were analyzed for 6 months. All the tests were carried out on a server having two Intel Xeon quad core processors with 48GB of RAM. The operating system was FreeBSD, and MTA server is qmail.

In this work, 105,211 spam and 51,299 ham emails were tested. These emails were offline tested by SA and our method SAN. The CPUs have 8 cores totally. Therefore,

eight emails were tested simultaneously. For an objective comparison, our software was run without any whitelist. In our test, 8.00 and 10.00 are selected as therequired_score values in SA, and the results are shown in Table 5.8.

Table 5.8 The comparison of SAN and SpamAssassin.

			SAN			SpamAssassin				
User	**Ham**	**Spam**	**FN %**	**FP %**	**Time (sec)**	**RS[1]=8.00**		**RS[1]=10.00**		**Time (sec)**
						FN %	**FP %**	**FN %**	**FP %**	
1	1929	215	6.05	0.31	0.031	15.35	0.47	20.93	0.00	6.13
2	2945	968	7.44	0.07	0.028	56.92	0.58	63.22	0.20	5.35
3	2818	210	43.81	0.11	0.030	24.29	0.57	40.95	0.11	5.88
4	2786	765	6.27	0.47	0.036	15.16	1.04	27.97	0.32	5.44
5	2150	1343	11.32	0.09	0.029	44.75	1.02	58.90	0.19	5.94
6	3853	6719	2.13	0.65	0.023	22.80	0.67	40.66	0.29	3.78
7	1661	1425	2.04	0.24	0.035	35.44	1.44	49.96	1.02	5.96
8	3276	1047	4.97	0.34	0.033	57.12	0.37	69.05	0.06	4.84
9	3082	1008	8.23	0.10	0.034	14.88	0.68	26.39	0.19	5.41
10	2091	1399	16.87	1.29	0.036	56.25	0.77	67.33	0.19	5.41
11	3915	1513	15.93	0.46	0.032	21.81	1.15	41.97	0.13	6.10
12	2299	1295	7.49	0.91	0.035	44.32	1.00	54.36	0.83	5.27
13	337	1034	3.97	1.19	0.031	44.58	0.00	52.42	0.00	5.83
14	3578	1216	18.01	0.31	0.032	26.89	0.64	44.90	0.34	5.26
15	1888	826	6.54	0.42	0.035	49.76	0.64	68.04	0.21	4.70
16	2939	721	8.60	0.00	0.035	8.88	1.26	21.22	0.41	4.34
17	2927	1648	13.05	0.20	0.033	41.93	0.65	56.55	0.38	4.34
18	3134	303	6.93	0.51	0.035	23.10	0.06	48.18	0.03	4.45
19	2804	1553	3.67	0.25	0.034	32.26	1.25	43.27	0.36	4.25
20	531	566	25.80	0.19	0.036	16.08	1.32	27.03	0.00	6.06
21	2981	896	14.17	0.40	0.030	42.08	0.87	61.05	0.47	4.99
22	1343	630	3.81	0.00	0.036	16.98	1.19	24.13	0.07	5.06
23	1022	1057	6.15	0.39	0.035	47.21	0.68	55.06	0.49	5.08
24	1971	900	6.89	0.41	0.032	33.78	1.12	45.33	0.66	5.13
25	3774	1128	14.27	0.45	0.028	43.97	0.90	60.55	0.58	5.68
26	3975	1400	9.43	0.40	0.036	30.50	1.06	48.36	0.13	4.54
27	3917	1667	16.98	0.00	0.035	27.17	1.20	46.97	0.82	4.70
28	1676	3504	5.59	0.18	0.036	29.82	0.60	38.67	0.12	4.70
29	4693	1851	4.97	0.15	0.033	27.66	0.55	41.71	0.13	4.23
30	3570	2638	8.53	0.11	0.034	9.06	0.70	21.57	0.45	7.47
31	5242	775	4.52	0.15	0.034	9.81	1.07	43.35	0.88	6.75
32	3753	2916	12.89	0.32	0.033	36.35	0.51	57.27	0.11	7.07
33	3975	2357	17.86	0.45	0.035	34.41	0.75	51.46	0.30	8.70
34	4085	1095	7.49	0.27	0.037	36.26	0.59	56.16	0.32	7.18
35	4514	1564	13.11	0.09	0.035	19.88	1.22	39.71	1.04	7.72
36	3777	1147	2.70	0.19	0.036	41.33	1.32	54.14	1.24	7.20
Avg			**8.773**	**0.287**	**0.029**	**27.115**	**0.712**	**39.734**	**0.311**	**4.785**
Total	**105,211**	**51,299**								

In Table 5.8, FP ratios of SAN and SA are similar when required_score is used as 10.0. SAN has advantages in FN ratios and time spent per email. In our tests, by decreasing the required_score of SA to 8.0, the FN values decreased to 27.115%, but that increased the FP ratio to 0.712%. Both scores are much lower in SAN. SA is not much effective on Turkish language, because it is content based spam filter.

5.5 The Weaknesses of the SAN

There is no ideal spam filtering system. Every spam filtering method has some weaknesses. Like many other methods, SAN has some weaknesses. We detected these weaknesses from the user feedbacks. We analyze these weaknesses in two groups.

5.5.1 The Weaknesses That Cause False-Positives

If a domain uses IP addresses for sending and receiving emails from different IP2 networks, then the emails, which are sent from an IP address outside of SAN, are treated as spam. The solution is adding IP2 and domain names for those emails to the whitelist. Emails, which are sent from a legitimate portal with private email addresses, are marked as spam. For example, some email groups, ScholarOne, Elsevier, are in this group. Because the count of these portals is very low, their IP addresses are also added to the whitelist.

5.5.2 The Weaknesses That Cause False-Negatives

Decreasing of FP emails has the risk of increasing of FN emails. These risks can be listed as follows:

• Two domains can send spam to each other’s domain name, if they are in the same IP2 network.

• Spammers can take email address from legitimate domains like Yahoo, Hotmail, Gmail. The method has no effect for those spam emails.

• Spammers can use hacked servers.

• Spammers can use open relay servers.

• Spammers can use a hacked account of legitimate server.

• Spammers can send spam emails with a newly purchased domain name.

These weaknesses can be fixed with content based filtering or with a reputation system as blacklisting. For the weakness mentioned in the last item, we proposed a method described in the next chapter.

CHAPTER SIX
DETECTING SPAM THROUGH THEIR SPF RECORDS

6.1 Overview

As mentioned in the fourth chapter, there are two deficiencies of the SPF implementation. The solution method of the first deficiency is described in the chapters 4 and 5. The solution of the second deficiency will be explained in this chapter. The second deficiency is explained as follows: Spammers create SPF records, by purchasing thousands of domain names. In order to block this type of spam, either content filtering that requires higher memory and CPU resources or a fast and dynamic blacklist technique is needed, which requires higher bandwidth.

Spam elimination based on content filtering requires a significant amount of processing power. The spammers are also aware of the checking; thus they purchase many domain names and then publish SPF records of domains, before sending spam. These types of spams are defined as SPF-bad (Esquivel et al., 2010). Before 2012, FN ratios were under 5%, but in 2012 it began to increase. After analyzing these types of spam, we discovered that the domain names had SPF records and the emails were sent from authorized IP addresses. These spams were tried to be eliminated with the black list. There are no known methods to block SPF-bad type of spam at pre-acceptance stage without blacklisting. It seems that, we need a new method for detecting spam which has SPF record. The purpose of this chapter is to filter out spam using DNS/SPF properties.

6.2 The Required Materials and Methods

In the proposed method, the following steps have to be followed:

1. The sending domain names are divided into bad-domains (spammers) and good-domains.
2. The DNS/SPF records of bad-domains (spammers) and good-domains are analyzed to identify their distinctive features.

3. The probability of occurrence (frequency) of distinctive features of good and bad-domain are calculated.
4. The calculated probabilities will be used as the initial values in Naïve Bayes classification to detect bad-domains.
5. The proposed method will be tested with real emails of volunteers.

To achieve all these steps, we need the SPF protocol structure and Bayes theorem, which are explained in the chapter 3.

6.3 Preparation of Test Data

We conducted two studies for preparation of test data with an interval of 1 year. First study was made in 2011 (Sipahi and Dalkılıç, 2012), and second 2012 (Sipahi and Dalkılıç, 2015). The data of the second study also includes the first data.

We used approximately 1 million emails in the first study; and 1.4 million emails in the second study. We used more than 1000 email addresses to collect test data. During the preparation of the test data, we took the following steps:

Step 1. The emails, which were sent with well-known domain names like gmail, hotmail, yahoo, facebook and twitter were removed from the test data in order to reduce process load. Such well-known domain names are in the ham domains group. Except for those, formal Turkish domain names such as banks and educational institutions were subjected to the same process. After removing them, 130,924 emails passed the SPF tests in the first study. However, in the second study, 1,004,687 emails passed the SPF tests.

Step 2. A content filtering was applied. The well-known tool SpamAssassin, which has a success rate of 95% in filtering spam, was used (SpamAssassin, 2012). SpamAssassin uses a variety of techniques including header and content analysis. In the last stage, we made manual control. At the end of this stage: The number of ham emails is 72,618 and the number of spam emails is 57,321 in the first study. In the

second study, the number of ham emails is 695,084 and the number of spam emails is 309,603.

Step 3. After the classification of emails as ham and spam, domain name classification started as ham-group and spam-group. For this purpose, we made manual control again. There is a risk of misclassification because of hacked accounts. In order to prevent this situation, a domain name would be written to the ham group, if at least one ham email was sent from the said domain name.

Most of the ham emails have been sent with a few domains in the ham group. Top 10 domains sent 148,004 (21% of total) emails; top 100 domains sent 380,327 (55% of total) emails, and top 500 domains sent 550,230 (79%) emails. Spammers can use legitimate domain names, by stealing passwords or creating bogus accounts from popular sites (hotmail, gmail, yahoo). Therefore, the domain names in both groups were discarded from the spam group. They belong to ham group only. Official domain names (e.g. government, universities) were transferred from the spam group to the ham group. The spam group consists of domains which only sent spam. To reduce errors, email sending domains which sent only one spam email were discarded.

After these studies, the domain names were grouped as ham and spam domains. The domain number in the ham group was 2315, and the domain number in the spam group was 3777 in the first study. In the second study, the domain number in the ham group was 10459, and the domain number in the spam group was 8310.

Looking at the number of emails, 72,618 emails were sent from domains in the ham group and 57,321 in spam group in the first study. In the second study, 909,626 emails were sent from domains in the ham group, and 95,061 in the spam group. The results of these studies are shown in the Table 6.1. The first column of Table 6.1 shows the steps of the preparation process. The second column shows the description. The subsequent columns show the numerical data of the studies.

Table 6.1 The categorization of test data per studies.

Steps	Sampled Entity	First Study	Second Study
1	email with SPF	130,924	1,004,687
2	ham emails	72,618	695,084
2	spam emails	57,321	309,603
3	ham domains	2,315	10,459
3	spam domains	3,777	8,310

6.4 The Characteristics of the SPF Records

The SPF protocol structure was described in the Section 3.3. The SPF records consist of the SPF version number, mechanisms, qualifiers (prefixes), and modifiers. The “all” mechanism is a decision-maker with its qualifiers (prefixes). Firstly, the prefixes of "all" mechanism were analyzed. Secondly, mechanisms and modifiers are analyzed.

6.4.1 The Analysis of the Qualifiers

The "all" mechanism should be used, if "redirect" modifier is not used. The results of the analysis of the first study are shown in Tables 6.2; and Table 6.3 is used for the second study. In both studies, similar rates are found with various number of domain names.

Table 6.2 The usage of prefixes of “all” in the first study.

Qualifiers	Ham	Ham %	Spam	Spam %
?all	279	13.25	176	4.97
~all	998	47.41	2889	81.22
-all	806	38.29	470	13.26
+all	22	1.05	9	0.25
Total	2105	100.00	3544	100.00

Table 6.3 The usage of prefixes of “all” in the second study.

Qualifiers	Ham	Ham %	Spam	Spam %
?all	1372	13.20	243	2.92
~all	4765	45.83	6475	77.93
-all	3958	38.07	1097	13.20
+all	130	1.25	55	0.66
none	171	1.64	439	5.28
Total	10396	100.00	8309	100.00

In Table 6.2 (first study), the row “none” is missing. Due to the small number of domain names, there was not such an SPF record missing the term “all”.

Two results in Table 6.2 and Table 6.3 stand out: First of all, the ratio of "-all" usage in ham-domain group is higher than the spam-domain group, approximately 38% to 13% in both studies. Only the policy "-all" is a necessary condition for rejecting the message from the recipient mail server. Therefore, it is recommended that the domain owners use the “-all” policy in their SPF. On the other hand, spammers want to keep the widest possible pool of IP addresses.

The second feature is the "~ all" usage. These ratios are 45.83% in ham group and 77.93% in spam group in the second study. In the first study these ratios are 47.41% to 81.22%. These ratios are similar too. An article that examined the network size of the SPF record and reached to a similar conclusion has also been presented (Mori et al., 2011).

6.4.2 The Analysis of the Mechanisms and Modifiers

We have made the second analysis in the first study in terms of simplicity and complexity. Some SPF records consist of only “a“, “mx” and “ptr”. We defined such records as simple. The other SPF records include “ip4:” “include”, “redirect”, etc. The complex SPF records require an effort, but simple SPF records do not. The results of the first study are shown in Table 6.4

Table 6.4 The usage of SPF terms as complexity in the first study.

Records type	Ham	Ham %	Spam	Spam%
Simple	314	13.56	2446	64.76
Complex	2001	86.44	1331	35.24
Total	**2315**	**100**	**3777**	**100**

Table 6.4 shows us that simple SPF records are widely used in spam-group (64% to 13%), while complex SPF records are used in the ham-group (86% to 35%). A detailed analysis of this usage of SPF terms is made in the second study. We have found the usage ratio of all mechanisms and modifiers. The results of the study about mechanisms and modifiers in SPF record are shown in Table 6.5. The colons ":" are for giving detailed description. For example, only "a" is used for the domain name itself, whereas to add another domain name "a:<hostname>" is used. Only the "mx" from domain is responsible for the use of email recipients, but "mx:<domain>" is used to specify a different domain name.

Table 6.5 The usage of SPF terms in the second study.

Mechanism	Ham	Ham %	Spam	Spam %
ip4	6612	63.60	1942	23.37
ip6	37	0.36	9	0.11
ptr	1073	10.32	5446	65.54
a	3599	34.62	6357	76.51
a:	1591	15.30	313	3.77
mx	5593	53.80	6667	80.24
mx:	1663	16.00	280	3.37
include	3452	33.21	601	7.23
redirect	179	1.72	26	0.31
exists	12	0.12	2	0.02
exp	6	0.06	2	0.02

The main feature shown in this table is that the simple mechanisms (a, mx, ptr) have a high rate in the spam group, but detailed mechanisms are used in the ham group. Spammers manage thousands of domains, and detailed SPF records take more time than simple SPF records. Therefore, spammers prefer to use simple SPF records.

In Table 6.5, the usage of the mechanisms “ptr”, “a” and “mx” is higher in the spam group. The “ptr” usage is 65.54% to 10.32%, the “a” usage is 76.51% to 34.62%, the “mx” usage is 80.24% to 53.80%. However, the usage of the mechanisms “ip4:”, “a:”, “mx:” and “include” is higher in the ham group. The “ip4:” usage is 63.60% to 23.37%; the “a:” usage is 15.30% to 3.77%; the “mx:” usage is 16.00% to 3.37%; and the “include” usage is 33.21% to 7.23%. These ratios have distinctive feature.

6.5 Implementation of Naïve Bayes on SPF Features

We mentioned Bayes’ Theorem and Naïve Bayes (NB) implementation in Section 3.5. We found distinctive features of SPF terms in ham and spam groups. We will combine both of them. The ratios in Tables 6.3 and 6.5 differ in ham and spam groups. These features will improve the efficiency of NB implementation. For this purpose, a software was developed in C programming language to implement NB. The pseudocode of this software is shown below:

```
Retrieve domain name from sender address
Resolve the SPF record of domain
Initial probability value is 1
foreach term in SPF record
   Calculate term ratio in ham group
   Find the ham result by multiplying the ratio and previous value
   Calculate term ratio in spam group
   Find the spam result by multiplying the ratio and previous value
Subtract ham value from spam value
if the result is positive
then domain is ham
   else domain is spam
end if.
```

Figure 6.1 Pseudo code of program.

The pseudocode above is executed for every domain in ham and spam groups. The FP and FN numbers of the domains are shown in Table 6.6. Two calculation examples are given below:

Ham example: domain1.com

SPF record: v=spf1 a mx ip4:64.145.94.0/24 –all

Probability of ham: P_{ham}= P(a)*P(mx)*P(ip4)*P(-all)

P_{ham} = 3599/10359 * 5593/10359 * 6612/10359 * 3958/10359 = 0.0457

P_{spam} = 6357/8309 * 6667/8309 * 1942/8309 * 1097/8309 = 0.0189

$P_{ham} > P_{spam}$

The result is ham.

Spam example: domain2.com

SPF record: v=spf1 a mx ptr ~all

P_{ham}= P(a)*P(mx)*P(ptr)*P(~all)

P_{ham} = 3599/10359 * 5593/10359 * 1073/10359 * 4765/10359 = 0.0089

P_{spam} = 6357/8309 * 6667/8309 * 5446/8309 * 6475/8309 = 0.3135

$P_{ham} < P_{spam}$

The result is spam.

Table 6.6 The results of the implementation.

Results	Ham group	Ham Ratio%	Spam group	Spam Ratio%
Ham results	8799	84.13	2553	30.72
Spam results	1660	15.87	5757	69.28

In summary, the results of the ham and spam groups can be explained as follows: The success rates are 84.13% in ham group (TrueNegative) and 69.28% in spam group (TruePositive). The FalsePositive (FP) rate is 15.87% and FalseNegative (FN) rates is 30.72%. After analyzing the domain names in FP and FN groups, the following interesting cases are detected: In FP domains, “com, tr, org, net, edu” TLDs are most visible. In FN domains, “in, com, br, net, info” TLDs are most visible. For this reason, domain name suffixes (TLD) for ham and spam groups were also analyzed.

6.5.1 Appending Domain TLDs to the Method

The TLDs are built up by ICANN (ICANN, 2015). The TLDs are divided into two groups as generic (gTLD) and country code (ccTLD). Their distribution is made considering to certain rules. Some TLDs are given (gov, edu, mil) to official institutions and some of them (com, net, org) are given to others. Most of the two-character suffixes (ccTLD) are given to countries. Each country has a different policy and pricing system. The distribution of suffixes in ham and spam groups is shown in Table 6.7.

Table 6.7 Distribution of TLDs per ham and spam groups.

TLD	HAM	HAM %	SPAM	SPAM %
com	5298	50.65	5362	64.52
tr	1474	14.09	8	0.10
net	1191	11.39	505	6.08
org	559	5.34	226	2.72
edu	265	2.53	3	0.04
uk	167	1.60	21	0.25
de	140	1.34	7	0.08
br	122	1.17	227	2.73
info	67	0.64	317	3.81
in	36	0.34	616	7.41
jp	24	0.23	96	1.16
ru	29	0.28	83	1.00

Table 6.7 shows TLDs, which are at least 1% in each group. Some TLDs such as “tr”, “uk”, “de” and “edu” have a higher ratio in the ham group. In contrast to that, info, br, in, jp and ru TLDs have higher ratios in the spam group. After these ratios are appended to the NB implementation software, it is evaluated again, and the results are shown in Table 6.8. The success rate (TrueNegative) increased from 84.13% to

87.63% after adding TLDs to the software in the ham group, and the success rate (TruePositive) also increased from 69.28% to 79.31% in the spam group.

Table 6.8 The results of the implementation after appending TLDs.

Results	Ham group	Ham ratio%	Spam group	Spam ratio%
Ham results	9165	87.63	1719	20.69
Spam results	1294	12.37	6591	79.31
Total	10459	100.00	8310	100.00

6.6 Live Test and Comparison with SpamAssassin

For a comparison, a server having two Intel Xeon processors with quad cores and 100 email addresses were used. We selected these addresses from those, who make frequently feedbacks as FN and FP. Their incoming emails were analyzed for 10 months.

Approximately 3 million (2,973,646) emails were classified by SAN. This method marked 1,847,898 emails as spam. The 740,816 of the remaining 1,125,748 emails were sent from the domains having SPF record. The software and SA were applied to these emails. After manually scanning the emails, 97,982 of them turned out to be spam emails which means the initial FN value is 97,892, and FN rate is 5.03%.

To minimize the FP rate, a whitelist was used. If the whitelist consists of 6 domains, FP ratio was 0.07%; for 10 domains, FP ratio was 0.06%, so 10 domains were preferred. In the SpamAssassin, the threshold (required_score) values were selected as 11 to minimize FP emails. Eight parallel processes were used in this test to reduce the execution time.

The parameters of the comparison are; total execution time, CPU load average, FN, FP. The results are summarized in Table 6.9.

Table 6.9 Summary of the comparison with SpamAssassin.

Comparison Criteria	**Our software**	**SA**	**First our software then SA**
Ham with SPF Record	740,816	740,816	740,816
Total execution times (min)	36	8901	8561
CPU load average	0.43	0.67	0.67
FN Counts	66,678	6082	6082
FP Counts	362	378	740
FP ratio %	0.06	0.06	0.12
FN ratio %	3.43	0.31	0.31

The average FP ratio of our software is 0.06% (whitelist with 10 domains), and the FN ratio is 3.43%. Nearly 1 of every 3 spam emails (31,214 of 97,892) is detected without content filtering.

In SpamAssassin, FP value is 0.06%, FN value is 0.31%. Based on the FN ratio, SA gives better results. However, based on the execution time, our software is better. SA takes 8901 minutes, while our software takes only 36 minutes. Both methods have advantages in different fields. We can combine both methods; first executing our software, then SA. Test results for the combined usage of our software and SA were shown in the last column of Table 6.9. In this way, as our software identifies 31,214 of the spam emails, that decreases the input to the SA to be 709,602 emails. Our software identifies that much spam in 36 minutes equivalent to the SA's 376 minutes. Our gain is 340 minutes. The test is for 100 users within 8 months; so, the gain will be 527 minutes in 12 months. The regular number of active users of the Dokuz Eylul University email system is 4300. For that much people, the gain will be 22,660 minutes that is approximately 16 days in 1 year. A gain of this size for a university is evidently not ignorable.

6.7 Comparison with Other Methods like Exposure

There are other studies to detect malicious DNS records. One of them is called "Exposure" (Bilge et al., 2014). It is a system that employs large-scale, passive DNS analysis techniques to detect domains that are involved in malicious activity. Exposure analyzes 14 features and the analysis results. A list of malicious domains is posted on a website (http://exposure.iseclab.org/), which can be used for comparison by the Internet community. The results of our study have been compared with the list on that website. The latest list given by Exposure is from the year 2012, which contains 42,097 malicious domains. Our list contains 8310 spam sending domains for the year 2012. The comparison of the two lists shows an intersection of 97 domain names, which means Exposure failed to identify 8213 spam sending domains. Comparison of Exposure's year 2012 list with our year 2013 list shows no intersection of domains. Therefore, Exposure's year 2012 list is incapable of identifying the new spam sending domains. However, 2012 and 2013 lists of our data have 74 common domains. The domain names have to be updated regularly, because spammers can change domain names frequently.

In their tests with previously unclassified domains, Kopis (Antonakakis et al., 2011) reached 0.53% FP ratios on the average. Exposure analyzed FP ratios for different feature sets and found the best value for 14 features to be greater than 0.4%. The FP ratios are not given in work (Hao et al., 2011). The results of the works above are much higher than our 0.06% FP value.

CHAPTER SEVEN
CONCLUSION

7.1 Conclusion

In this thesis, we have developed and implemented two different methods to filter out spam emails. The first method is Sender Authentication Network, which predicts SPF record of domains having no SPF record. This method aims to fix the first deficiencies in the SPF protocol. In the beginning, we had some hesitation; since the implementation of Sender Authentication Network method had some risks. We expanded the authorized IP addresses in the SPF record. Although an SPF record may have only one IP address, IP address is extended to all the IP addresses having the same first 2 octets (over 65000). This expansion of the authorized IP addresses may cause the reduction of FP emails, while increasing FN emails. As a result of seven-years of study, few FNs (less than 10) that support this worry, were reported by the users.

We also compared our method SAN with SpamAssassin (SA). As a result of this comparison, it is found out that SAN is very fast, having nearly the same FP ratios. SAN spent 0.029 seconds per email, and SA spent 4.785 seconds per email. SAN can check 168 emails while SA checks only one email in the same amount of time.

SAN is simple, it was tested with the live emails of the Dokuz Eylul University domain, and it runs quickly; FN and FP ratios are acceptable. SAN can run without any initial table, and does not need extra computational resources. Therefore, it is a good candidate for being a spam filter.

Spammers are using the SPF records of newly purchased domain names to send spam emails. Detecting malicious SPF records without content filtering is an important gain. We proposed a new method, which analyzed the terms in the SPF record of malicious and legitimated domain names. Our second method saves time and hardware costs in the contest against spammers, who invent new tactics every day. Our second

method contributes to spam filtering techniques through analyzing daily, live email; rather than the traditional methods based on out-of-date datasets. The second method analyzes malicious domains by their detailed SPF properties and has better FP results (0.06%) than those of the IP reputation systems (>0.4%). The second method can help reduce the content-filtering processing time of 4300 email users by 3.82%, which is equivalent to 16 days of SpamAssassin's running time within a year.

7.2 Future Works

As for future work, we plan to work on two points: First, reducing FP emails. For this purpose, the email headers sent outside of the SAN can be analyzed. Secondly, some relations among the IP address, domain name and header information of the email can be investigated to expand the SAN. Some domain names can belong to the same organization. Finding the relationship between them helps to reduce FP emails. For this aim, email header information or disclaimer can give us a hint.

In the second method, we used Naïve Bayes (NB) classifier implementation. NB is one of the machine learning methods. Further research on using machine learning methods on SPF records; other than NB, is presently pursued by the authors for further improvement.

REFERENCES

Antonakakis, M., Perdisci, R., Lee, W., Vasiloglou, N., & Dagon, D. (2011). Detecting malware domains at the upper DNS hierarchy. *Proceedings of the 20th USENIX Conference on Security,* 27.

Allman, E., Callas, J., Delany, M., Libbey, M., Fenton, J., & Thomas, M. (2007). *DomainKeys Identified Mail (DKIM) Signatures*, RFC 4871.

Arora, H. (2013). *How email works? – email basic concepts explained.* Retrieved October 1, 2016, from http://www.thegeekstuff.com/2013/05/how-email-works/.

Bernstein, D. J. (2007). *The qmail security guarantee.* Retrieved May 26, 2016, from https://cr.yp.to/qmail/guarantee.html.

Bilge, L., Sen, S., Balzarotti, D., Kirda, E., & Kruegel, C. (2014). EXPOSURE: a passive DNS analysis service to detect and report malicious domains. *ACM Transactions on Information and System Security (TISSEC), 16* (4), 1-28.

Carpinter, J., & Hunt, R. (2006). Tightening the net: A review of current and next generation spam filtering tools. *Computers & Security, 25* (8), 566–578.

Carreras, X., & Marquez, L. (2001). Boosting trees for anti-spam email filtering. *In 4th International Conference on Recent Advances in Natural Language Processing (RANLP-2001),* 58–64.

Chhabra, S., Yerazunis, W. S., & Siefkes C. (2004). SpamFiltering using a markov random field model with variable weighting schemas. *In Proceedings of the 4th IEEE International Conference on Data Mining,* 347–350.

Cook, D.L., Gurbani, V. K., & Daniluk, M. (2008). Phishwish: A simple and stateless phishing filter. *Security and Communication Networks, 2,* 29-43.

Cunningham, P., & Delany, S. J. (2007). k-Nearest neighbour classifiers. *Technical Report UCD-CSI-2007-4, Dublin: Artificial Intelligence Group*. Retrieved May 3, 2016, from https://csiweb.ucd.ie/files/UCD-CSI-2007-4.pdf.

Dalkılıç, G., Sipahi, D., & Özcanhan, M.H., (2009). A simple yet effective spam blocking method. *SIN '09 Proceedings of the 2nd International Conference on Security of Information and Networks*, 179-185.

Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., & Samarati P. (2004). P2P-based collaborative spam detection and filtering. *Proceedings of the Fourth International Conference on Peer-to-Peer Computing, IEEE Computer Society,* 176–183.

Danisch, H. (2002). *Reverse MX*. Retrieved August 10, 2016, from http://en.citizendium.org/wiki/Reverse_MX.

Drucker, H., Wu, D., & Vapnik, V. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural Networks, 10*, 1048-1054.

Eklund C. (2004). Spam - from nuisance to Internet infestation. R. Kantola (Ed.), *Peer to peer and SPAM in the Internet* (126-134). Espoo: Teknillinen korkeakoulu.

EDN Network (2015). *1st spam e-mail is sent, May 3, 1978*. Retrieved May 4, 2016, from http://www.edn.com/electronics-blogs/edn-moments/4413320/1st-spam-e-mail-is-sent--May-3--1978.

Edwards, J. (2016). *IT Security: The real cost of spam.* Retrieved May 4, 2016, from http://www.itsecurity.com/features/real-cost-of-spam-121007/.

Emailtray (2012). *Email spam trends at a glance: 2001 - 2012*. Retrieved May 14, 2016, from https://www.emailtray.com/blog/email-spam-trends-2001-2012/.

Esquivel, H., Akella, A., & Mori, T. (2010). On the effectiveness of IP reputation for spam filtering. *Proceedings of the 2nd International Conference on communication systems and networks,* 40-49.

Fecyk, G. (2003). *Designated Mailers Protocol*. Retrieved June 19, 2016, from http://www.potaroo.net/ietf/all-ids/draft-fecyk-dsprotocol-03.txt.

Fumera, G., Pillai, I., & Roli F.,(2006). Spam filtering based on the analysis of text information embedded into images. *The Journal of Machine Learning Research, 7,* 2699-2720.

Gargiulo, F., & Sansone, C. (2008). Combining visual and textual features for filtering spam emails. *International Conference on Pattern Recognition,* 1-4.

Gomes, L. H., Cazita, C., Almeida, J., Almeida, V., & Meira, W. (2004). Characterizing a spam traffic. *IMC '04: Proceedings of the Fourth ACM SIGCOMM Conference on Internet Measurement, ACM Press,* 356–369.

Gómez Hidalgo, J. M., Maña López, M. & Puertas Sanz, E. (2000). Combining text and heuristics for costsensitive spam filtering. *Fourth Computational Natural Language Learning Workshop, Lisbon, Portugal,* 99-102.

Gonzáles-Tavalán, G. (2006). A simple, configurable SMTP antispam filter: Greylists. *Computers & Security, 25,* 229-236.

Hafiz, M., Johnson, R. E., & Afandi, R. (2004). The security architecture of qmail. *In Proceedings of the 11th Conference on Patterns Language of Programming (PLoP'04)*. Retrieved August 12, 2016, from http://hillside.net/plop/2004/papers/mhafiz1/PLoP2004_mhafiz1_0.pdf.

Hao, S., Feamster, N., & Pandrangi, R. (2011). Monitoring the initial DNS behavior of malicious domains. *In Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference,* 269-278.

Harris, E. (2003). *The next step in the spam control war: Greylisting*. Retrieved July 12, 2016, from http://projects.puremagic.com/greylisting/whitepaper.html.

Herzberg, A. (2009). DNS-based email sender authentication mechanisms: A critical review. *Computers & Security 28* (8), 731-742.

ICANN, (2015). *Internet corporation for assigned names and numbers | ICANN*. Retrieved August 12, 2015, from http://www.icann.org.

Joachims, T. (1997). A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization. *Proceedings of the Fourteenth International Conference on Machine Learning*, 143-151.

Jung, J., & Sit, E. (2004). An empirical study of spam traffic and the use of DNS black lists. *In Proceedings of ACM SIGCOMM Internet Measurement Conference (ICM'04). Taormina, Italy*, 370–375.

Koski, T., & Noble, J. (Eds.). (2009). *Bayesian networks: an introduction*. Padstow: Wiley Publishing.

Levine, J. (2010). DNS blacklists and whitelists. *IRTF anti-spam research group.* Retrieved May 4, 2016, from https://tools.ietf.org/html/rfc5782.

LISTSERV, (1995). *History of LISTSERV*. Retrieved April 14, 2016, from http://www.lsoft.com /corporate/history-listserv.asp.

Lyon, J., & Wong, M. (2006). *Sender ID: Authenticating E-Mail*, RFC 4406, 2006.

Maildir, (1998). *Man pages for qmail 1.03: maildir(5).* Retrived May 18, 2016, from http://www.qmail.org/qmail-manual-html/man5/maildir.html.

Mangalindan, M. (13 November 2002). Spam queen: for bulk e-mailer, pestering millions offers path to profit. *Wall Street Journal.*

Mori, T., Takahashi, Y., Sato, K., & Ishibashi K. (2011). How is e-mail sender authentication used and misused? *In 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference,* 31-37.

Nucleus research (2015). Retrieved June 11, 2016, from https://nucleusresearch.com/press/2nd-annual-spam-report-cost-of-spam-more-than-doubled-in-past-year-to-1934-annually-per-employee/.

O'Brien, C., & Vogel, C. (2003). Spam filters: bayes vs. chi-squared; letters vs. words. *In: Proceedings of the 1st International Symposium on Information and Communication Technologies, ISICT '03,* 291–296.

Oppermann, A. (1998). *The big qmail picture.* Retrieved May 3, 2016, from http://www.nrg4u.com/qmail/the-big-qmail-picture-103-p1.gif.

Ostrihon, P., & Rajabiun, R. (2008). The robustness of new email identification standards. *Virus Bulletin Conference October 2008*, Retrieved July 4, 2016, from http://www.comdomsoft.com/files/Ostrihon-Rajabiun-VB2008.pdf.

Pantel, P., & Lin, D. (1998). SpamCop: A spam classification and organization program. *AAAI Technical Report WS-98-05.* 95–98. Retrieved May 31, 2016, from http://www.aaai.org/Papers/Workshops/1998/WS-98-05/WS98-05-17.pdf.

RFC 822, (1982). *Standard for ARPA Internet text messages*. Retrieved May 3, 2016, from https://www.ietf.org/rfc/rfc0822.txt.

RFC 2045, (1996). *Multipurpose Internet mail extensions*. Retrieved May 4, 2016, from https://www.ietf.org/rfc/rfc2045.txt.

RFC 4406, (2006). *Sender ID: Authenticating e-mail*. Retrieved May 6, 2016, from https://tools.ietf.org/html/rfc4406.

RFC 4408, (2006), *Sender policy framework (SPF) for authorizing use of domains in e-mail.* Retrieved May 8, 2016, from https://tools.ietf.org/html/rfc4408.

RFC 4632, (2006). *Classless inter-domain routing (CIDR): The Internet address assignment and aggregation plan*. Retrieved May 9, 2016, from https://tools.ietf.org/html/rfc4632.

RFC 4880, (2007). *OpenPGP message format*. Retrieved May 12, 2016, https://www.ietf.org/rfc/rfc4880.txt.

RFC 5617, (2009). *DomainKeys identified mail author domain signing practices (ADSP).* Retrieved May 15, 2016, from https://tools.ietf.org/html/rfc5617.

RFC 5518, (2009). *Vouch by reference*. Retrieved May 17, 2016, from https://tools.ietf.org/html/rfc5518.

RFC 5751, (2010). *Secure/multipurpose Internet mail extensions (S/MIME) version 3.2 message specification*. Retrieved May 22, 2016, from https://tools.ietf.org/html/rfc5751.

RFC 5585, (2009). *DomainKeys identified mail (DKIM) service overview*. Retrieved May 23, 2016, from https://tools.ietf.org/html/rfc5585.

RFC 5782, (2010). *DNS blacklists and whitelists*. Retrieved May 25, 2016, from https://tools.ietf.org/html/rfc5782.

RFC 7489, (2015). *Domain-based message authentication, reporting, and conformance (DMARC).* Retrieved May 27, 2016, from https://tools.ietf.org/html/rfc7489.

RFC 7530, (2015). *Network file system (NFS) version 4 protocol.* Retrieved May 28, 2016, from https://tools.ietf.org/html/rfc7530.

SMTP Mail Transaction Process, (2005). Retrieved May 29, 2016, http://www.tcpipguide.com/free/t_SMTPMailTransactionProcess-3.htm.

Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. *AAAI-98 Workshop on Learning for Text Categorization,* 55-62.

Sinha, S., Bailey, M., & Jahanian, F. (2008). Shades of grey: On the effectiveness of reputation-based blacklists. *In 3rd International Conference on MALWARE,* 57-64.

Sipahi, D., & Dalkılıç, G. (2012). Determination of SPF records for the intention of sending spam. *In IEEE Signal Processing and Communications Applications Conference (SIU),* Retrieved May 12, 2013, from http://dx.doi.org/10.1109/SIU.2012.6204709.

Sipahi, D., Dalkiliç, G., & Ozcanhan, M. H. (2015). Detecting spam through their Sender Policy Framework records. *Security and Communication Networks, 8* (18), 3555–3563.

Soonthornphisaj, N., Chaikulseriwat, K., & Tang-On P. (2002). Anti-spam filtering: a centroid-based classification approach. *Signal Processing, 2002 6th International Conference on.* 1096–1099.

Soucy, P., & Mineau, G. W. (2001). A simple k-NN algorithm for text categorization. *IEEE International Conference on Data Mining (ICDM'01), 28* 647-648.

SpamAssassin, (2004). Retrieved April 12, 2016, from http://spamassassin.apache.org/

SpamAssassin, (2012). Retrieved May 23, 2016, from http://svn.apache.org/repos/asf/spamassassin/branches/3.3/README.

Spamhaus, (2016). *The definition of spam*. Retrieved on January 24, 2016, from https://www.spamhaus.org/consumer/definition/.

Stats, (2015). Retrieved May 22, 2015, from http://spf-all.com/stats.html.

SPF best-guess, (2010). *FAQ/Best guess record. What is best-guess?* Retrieved June 12, 2016, from http://www.openspf.org/FAQ/Best_guess_record.

Symantec Intelligence Reports, Archives, (2016). Retrieved May 11, 2016, from https://www.symantec.com/security_response/publications/archives.jsp.

Taylor, B. (2006). Sender reputation in a large webmail service. *Third Conference on Email and Anti-Spam (CEAS 2006)*. Retrieved June 13, 2016, from https://research.google.com/pubs/archive/45.pdf.

The Museum of Email (2011). *First Spam Sent in 1978.* Retrieved June 25, 2016, from http://email-museum.com/2011/10/06/first-spam-was-in-1978/.

Tukey, J.W. (1977). *Exploratory data analysis.* Boston: Addison-Wesley.

Turk Telekom, (2009). *Türkiye' de e-postalar spam'den temizleniyor.* Retrieved June 3, 2016, from https://kurumsal.turktelekom.com.tr/internet/web/destek/bilgi-bankasi/sayfalar/turkiye-de-e-postalar-spamden-temizleniyor.aspx.

Twining, R. D., Willianson, M. M., Mowbray, M., & Rahmouni, M. (2004). Email prioritization: Reducing delays on legitimate mail caused by junk mail. *Proceedings Usenix Annual Technical Conference, Boston, MA.* Retrieved June 15, 2016, from http://www.hpl.hp.com/techreports/2004/HPL-2004-5.pdf.

Wong, M. & Schlitt, W. (2006). *Sender policy framework (SPF) for authorizing use of domains in e-mail, version 1*. Retrieved March 21, 2016, from https://www.ietf.org/rfc/rfc4408.txt.

WIDE Project, (2012). *Measurement results on deployment ratio of domain authentications.* Retrieved June 15, 2016, from http://member.wide.ad.jp/wg/antispam/stats/index.html.

Zimmermann, P. R. (1991). *Creator of PGP and co-founder of silent circle*. Retrieved June 15, 2016, from http://www.philzimmermann.com/EN/background/index.html.