Evaluation of Anti-spam Method Combining Bayesian Filtering and Strong Challenge and Response

Manabu IWANAGA†, Toshihiro TABATA‡, and Kouichi SAKURAI‡
Kyushu University
†Graduate School of Information Science and Electrical Engineering
‡Faculty of Information Science and Electrical Engineering
6-10-1 Hakozaki, Higashi-ku
Fukuoka, Fukuoka 812-8581 Japan
iwanaga@itslab.csce.kyushu-u.ac.jp, {tabata, sakurai}@csce.kyushu-u.ac.jp

Abstract

Recently, various schemes against spam are proposed because of rapid increasing of spam. Some schemes are based on sender whitelisting with auto registration, a principle that a recipient reads only messages from senders who are registered by the recipient, and a sender have to perform some procedure to be registered (challenge-response.) In these schemes, some exceptions are required to show error mail to a sender of an original message. However, spammers can abuse this exception to send spam to users. We have proposed improved scheme in [1], combining challenge-response and Bayesian filtering. In this paper, we make tests on our scheme and a scheme using only Bayesian filtering to show efficiency of our scheme.

Keywords: Network Security, spam, Bayesian filtering, Challenge-response

1 Introduction

According to popularization of email, spam is increasing because it is very cheap way to advertising. As users take measure to avoid spam, spammers also increase their sophistication of spamming, e.g. they fake headers of email including sender's address, and they tend to avoid using frank words which can be filtered with simple blacklist of words. According to some latest researches, approximately half of all email received by workers are spam. Massive spam filling users' mailbox not only irritates users, but makes it painful work for users to pick out messages users really need to read, especially for users who constantly send and receive messages to/from a large number of people.

Some schemes are based on a principle that a recipient reads only messages from senders registered in recipient's list. In these schemes, some exceptions are required for users to read error mail (bounce message), since it is impossible for recipients to add all possible senders of error mail, mailer daemon, to his sender-list. However, spammers can abuse this exception to send spam to users. Disguising their spam as error mail, spammers can show their spam to recipients using these schemes. In addition, if we have to consider a threat of wiretapping, the situation becomes worse, that is to say, if spam contains a copy of messages you sent, it is fairly hard for your computer to find out that it is spam.

In this paper, we explain modified scheme we proposed in [1], using not only challenge-response but also Bayesian filtering, to inspect error mail. Our proposed scheme applies Bayesian filtering to error mail, and applies the challenge-response scheme proposed by M. Jakobsson et al. [2] to all messages except error mail. Using this scheme, we can avoid spam disguised as error mail according to words in message-body and header, assuring certain receipt of legitimate messages from registered sender.

2 Related Work

To avoid spam, various schemes are proposed and used. Common strategies used in these schemes are classified into following viewpoint.

- Watching behavior of SMTP connection (e.g. Greylisting [3])
- Blacklisting IP addresses used to send spam (e.g. ORDB [4])
- Verification of source domain with DNS (e.g. RMX [5])
- Contents filtering (e.g. Spamassassin [6] and Bayesian filter [7, 8])
- Whitelisting of sender with auto-registration (e.g. [9, 10])

Watching behavior of SMTP connection, Blacklisting IP addresses and Verification of source domain are mainly used by mail servers. It is pretty hard for users to adopt on their PCs, because a mail server can omit some information of sender when it delivers to users' maildrop. On the other hand, users can

adopt contents filtering and whitelisting of sender easier, without modifying servers and SMTP (needless to say, these methods can be adopted on a server.)

Nowadays, Bayesian filtering is becoming very popular approach among users. Bayesian filtering uses statistical approach for detecting spam, and it has become famous last year with [7] by P. Graham. Bayesian filter calculates probabilities that the mail containing this word is spam for any words, according to past mail. Furthermore, Bayesian filter adds its knowledge about spam and non-spam from new mail, according to judgement of Bayesian filter itself. Therefore a user only has to teach his/her filter in case his/her filter makes a mistake. Even if spammers use obfuscated words, Bayesian filter also learns these words, then obfuscated words are used as obvious evidence. There are many implementations available [11, 12, 13].

2.1 Whitelisting of sender

"Whitelisting of sender" means a principle that a recipient reads only messages from senders registered by the recipient. This approach aims at killing off send and forget by spammers, or force up cost for sending spam. Legitimate senders have to perform the procedure once for each new recipient (challenge-response, in some scheme one procedure is valid for all recipients in same domain.) Additionally, in challenge-response scheme challenge is sent to an address written in "from", therefore these schemes also aim at preventing sender address forging.

In schemes of challenge-response, senders have to show evidence that they are legitimate senders who are registered. However, a spammer can wiretap the evidence, because email is transferred in plaintext among domains. Especially in some schemes an adversary only has to listen in "from" and "to" in the header of messages to break these schemes.

To avoid this, the scheme by M. Jakobsson et al. [2] adopts a manner that a recipient grants cryptographic key to legitimate senders. Their scheme uses Message Authentication Code (MAC). In their scheme, adversaries (spammers) are defined as active. That is to say, they not only wiretap communication channel, but remove and/or inject any messages at will. Therefore, this scheme adopted MAC calculated from each message to defeat wiretapping, and only a sender and a recipient know a key of MAC. A sender makes a setup message to obtain a key. The message contains a proof that the sender has performed a certain computational task or a monetary expense. Jakobsson's scheme can also detect that a legitimate message incoming is altered to spam by spammer.

3 Spam disguised as error mail

However, user must consider error mail, which does not have MAC. Since error mail is essential to notify failure to a sender, it must not be eliminated from user's screen. On the other hand, spammers may attempt to disguise their spam as error mail. The spam is disguised as error mail and escapes from spam-protection, but a user who tries to read it encounters malicious spam.

When a user's computer receives spam in a form of error mail, following situations are possible:

1. A spammer impersonates the user as a source of spam, then undeliverable spam was bounced back to the user's computer with error message according the header of the spam. This case can be divided into two cases by spammer's intention:

   (a) The spammer wants to hide himself/herself, so he impersonates the user. Spammers tend to hide their own address to avoid escape that their spam is filtered by their addresses, and to hinder being reported to his/her Internet Service Provider (ISP). For that reason, they often assume non-existent address, and in some case they personate other person's address. As a result, a personated recipient receives large amount of error mail.

   (b) It is spammer's primary purpose to bounce spam back to the user, so he sends spam to non-existent address intentionally. The spammer aims at that his/her spam attracts recipients' attention, or recipients' and servers' filters may react differently from behavior for common spam.

2. A spammer disguises spam as error mail, and sends it to the recipient directly. It is done by the same reason as (1b) and spammers can include their advertisement in any part of messages. If a spammer can eavesdrop on legitimate messages and makes his/her spam disguised as error mail from the legitimate messages, this way may be a serious threat.

A user can add some tags to one's messages to recognize whether error mail is for user's messages, then ignore error mail which is not correspond to one's messages (case 1.) It is safer if tags are encrypted and nobody can make the tags other than the user. A user can also achieve the same purpose by recording outbound messages to check an error mail with them.

However, neither message body nor a communication channel is encrypted. If a spammer can wiretap users' messages, the spammer can disguise spam as error mail whose original messages is one the recipient has sent. In this case, it is harder for recipients'

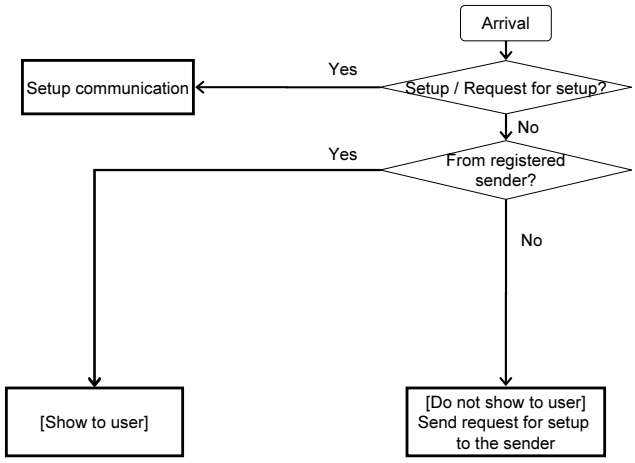

Figure 1. Process flow in [2]

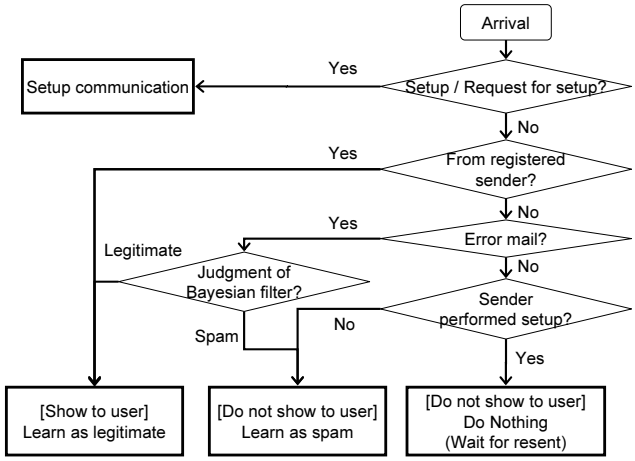

Figure 2. Process flow in our scheme

computers distinguish real error mail from disguised spam. So is the scheme [2]. Therefore additional methods against disguised spam are required to distinguish these messages.

Error mail notifies a sender that a message has a trouble with delivering. Spam filters must pass error mail. Since challenge-response cannot be applied to messages like error mail, we treat error mail with another principle. However, spammers can disguise their spam as legitimate error mail. We should not pass spam disguised as error mail.

Briefly, there are two manners for creating disguised spam. First, spammers can disguise origin of their spam with anyone else, and the spam can be sent back to a user who is impersonated. Second, spammers can send forged error mail. It becomes harder to prevent spam, if spammers eavesdrop on communication channels and forge error mail from eavesdropped messages.

4 Our Proposed Scheme

We proposed improved scheme at [1], based on Jakobsson's [2], aiming at preventing disguised spam from shown to recipient as error mail. In this scheme, we use Bayesian filtering [7, 8] as a to prevent disguised spam. In our scheme, Bayesian filter only makes judgements to distinguish error mail from disguised spam, i.e. non-

error mail (legitimate messages and spam, which do not seems error mail) are only learned by Bayesian filter, not judged. This scheme can keep using challenge-response and prevent spam that is disguised as error mail from MTAs, using a Bayesian filter.

In our Scheme, a message is processed with following rules.

1. If a message has a valid MAC, then the message is regarded as legitimate one and the Bayesian filter learns the message as a legitimate message, then the message is shown to recipient.

2. If a message is a request for setup or a setup communication that contains evidence of accomplishment of computational task, then a setup is performed automatically.

3. If a message seems an error mail, then the Bayesian filter judges whether the message is legitimate or not.

   (a) If the Bayesian filter judges that it is legitimate, the message is learned as a legitimate message, and shown to recipient.

   (b) If the Bayesian filter judges that it is spam, the message is learned as a spam, and isolated.

4. If a message does not meet all above conditions, then the message is isolated and is regarded as one from an unregistered sender. Recipient's computer sends a request to the sender for setup. However, this sender may have legitimate purpose to send a message, or of course he/she may have malicious intent, so learning is not yet.

   (a) If the sender does not perform a setup within the certain period, the message is considered to be sent for malicious intent. So the Bayesian filter learns the message as a spam.

   (b) If the sender does, the message is considered to be sent for legitimate purpose. Though, the Bayesian filter does not learn the message, because the schemes [2, 1] says the message should be resent in this case to prevent altering, so the scheme waits for the message resent.

Figure 1. shows the original scheme [2], and figure 2. shows the flow of our scheme.

4.1 Discussion

Our scheme fills the hole in [2] with Bayesian filtering, so our scheme obviously works better than [2]. Since many spammers forge sender's address on their spam to avoid responsibility for their spam nowadays, it is important to prevent disguised spam from shown to a user.

On the other hand, there is a question, "Why do not you use only Bayesian filtering?" While our proposed scheme is bother (in schemes [2, 1], senders are assumed to install software and perform setup) for legitimate senders. Furthermore, our scheme is more complex than a scheme using only Bayesian filtering. If our scheme is not more accurate than a scheme which uses only Bayesian filter, there is little advantage to our scheme. Therefore we made tests on simple Bayesian filter and our scheme to examine its accuracy (average number of mistakes) by simulating legitimate non-error messages, legitimate error mail, obvious spam and disguised spam.

In the tests, we simplified the problem on senders who does not respond to request for setup, because of simplicity of tests. In our simplified view, some percentage of legitimate messages are regarded as spam, because of lack of response, and the other legitimate messages are correctly processed. None of spammer is assumed that he/she performs setup, therefore all non-error spam is regarded as spam. Additionally, the case of message from unregistered sender which he/she accord to perform setup is regarded similarly as message from registered sender.

5 Tests

In the tests, we performed following process.

1. Take out some number of legitimate messages and spam, and make Bayesian filter learn this messages with specific designation (legitimate or spam.)

2. Mix and rearrange the rest legitimate messages and spam randomly.

3. Process with each scheme (simple Bayesian filtering / our scheme) one by one.

In case of simple Bayesian filtering, Bayesian filter always judges whether the message is spam, and learns the message according to judgement by itself. In case of our scheme, following process is performed with applying the simplification.

1. If a message is a request for setup or a setup communication that contains evidence of accomplishment of computational task, then a setup is performed automatically.

2. If a message is actually legitimate non-error message,

   (a) By some percentage, a sender is assumed that he/she does not perform setup, therefore the message is learned as spam by Bayesian filter.

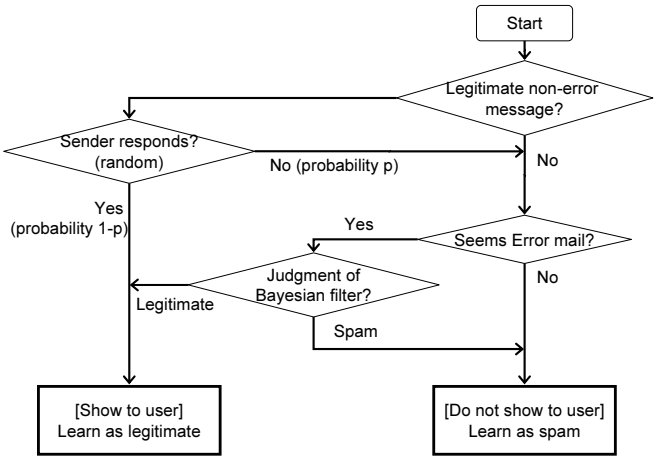

Figure 3. Simplified process flow in tests

(b) Otherwise, a sender is assumed that he/she perform setup, therefore the message is learned as legitimate by Bayesian filter.

3. If a message is actually legitimate error mail or disguised spam, since both of them seem error mail equally, a Bayesian filter judges whether the message is legitimate or not.
   (a) If the Bayesian filter judges that it is legitimate, the message is learned as a legitimate message.
   (b) If the Bayesian filter judges that it is spam, the message is learned as a spam.
4. If a message does not meet above conditions, the message is non-error spam. Since spammer is assumed that he/she never perform setup, the message is learned as spam.

In the tests, we used bsfilter [11], one of implementations using ruby [14]. Bsfilter has three choices for spam probability for each word and combined probability for mail: Paul Graham method, Gary Robinson method, Gary Robinson-Fisher method. We used Paul Graham method, and we use 0.9 for threshold, standard rate for Paul Graham method. We had 773 legitimate messages and 176 spam, most of them have subjects and bodies written in Japanese. Bsfilter supports handling Japanese messages with bigram. While we can point out and correct mistakes by Bayesian filter if it makes mistakes, we does not do correction.

6 Result

We show the result on table 1. There were too many false-negatives throughout the tests. It seems because the number of spam messages we had was too small.

First we discuss with a gap between simple Bayesian filtering scheme and our scheme without any unresponding senders. Combining challenge-response and Bayesian filtering, we reduced false-negatives for disguised spam, without increase of false-positives for legitimate error mail.

Second, we consider the relationship between percentage of unresponding senders and error rates. When there are some senders who do not respond for request for setup, false negatives are decreasing and some false positives happen. It seems because probabilities of only-legitimate words increase, and only-spam words tend to be preceded in calculating. On the whole, our scheme processes incoming messages a little better, combining challenge-response and Bayesian filtering.

7 Conclusion

In this paper, we have explained our scheme proposed in [1], then made tests with simple Bayesian filtering scheme and our scheme. Our proposed scheme adopts Bayesian filtering to distinguish error mail from spam disguised as error mail. Our scheme protects user from not only simple sender impersonation but also false error mail with eavesdropping. According to the tests, our scheme distinguishes legitimate error messages from disguised spam a little better than simple Bayesian filtering scheme.

In order for challenge-response to work well, we require suitable method for challenge-response that it is not annoying for legitimate senders. And again, not only error mail processing but our proposed scheme rely on sender's address in a message, so if an adversary (not only a spammer but a DoS attacker) personates other's address, although adversary cannot send

Table 1. False positives and false negatives

Initial Learning	Method	Unresponse Rate p(%)	Average of number of misjudge (probability)			
			Condition A		Condition B	
			False positives	False negatives	False positives	False negatives
$\frac{1}{2}$	Bayesian only		0.0 (0.00%)	**4.2 (14.94%)**	0.0 (0.00%)	**3.8 (13.69%)**
	Our scheme	0	0.0 (0.00%)	**3.2 (11.43%)**	0.0 (0.00%)	**2.9 (10.36%)**
		10	0.0 (0.00%)	2.8 (10.00%)	0.0 (0.00%)	2.4 (8.57%)
		20	0.0 (0.00%)	1.8 (6.43%)	0.0 (0.00%)	1.8 (6.43%)
		30	0.0 (0.00%)	1.8 (6.43%)	0.0 (0.00%)	0.8 (2.86%)
		40	0.1 (0.46%)	1.5 (5.36%)	0.2 (2.00%)	0.7 (2.50%)
		50	0.2 (0.91%)	1.0 (3.57%)	0.2 (2.00%)	1.0 (3.57%)
$\frac{1}{5}$	Bayesian only		0.0 (0.00%)	**17.5 (39.00%)**	0.0 (0.00%)	**17.6 (39.15%)**
	Our scheme	0	0.0 (0.00%)	**11.4 (25.33%)**	0.0 (0.00%)	**9.1 (20.22%)**
		10	0.0 (0.00%)	6.8 (15.11%)	0.0 (0.00%)	5.8 (12.89%)
		20	0.0 (0.00%)	6.2 (13.78%)	0.4 (2.50%)	3.2 (7.11%)
		30	0.0 (0.00%)	3.2 (7.11%)	0.5 (3.13%)	2.2 (4.89%)
		40	0.3 (0.88%)	4.2 (9.33%)	0.4 (2.50%)	3.5 (7.78%)
		50	0.2 (0.59%)	3.0 (6.67%)	1.0 (6.25%)	4.4 (9.78%)

spam itself, personated person is annoyed with wrong request for a setup in our proposed method. To avoid this, we require a way to prevent personation. Preventing personation has a significant role in anti-spam.

Acknowledgement

This research is partly supported by the 21st Century COE Program "Reconstruction of Social Infrastructure Related to Information Science and Electrical Engineering."

References

[1] M. Iwanaga, T. Tabata, K. Sakurai, Preventing Spam Disguised as Error Mail, *International Symposium on Information Science and Electrical Engineering*, Fukuoka, Japan, 2003.

[2] M. Jakobsson, J. Linn, J. Algesheimer, "How to Protect Against a Militant Spammer," *Cryptology ePrint archive*, report 2003/071, 2003.

[3] E. Harris, The Next Step in the Spam Control War: Greylisting, 2003, http://projects.puremagic.com/greylisting/.

[4] Open relay database, http://www.ordb.org/.

[5] The RMX DNS RR and method for lightweight SMTP sender authorization, *Internet-Draft*, http://www.ietf.org/internet-drafts/draft-danisch-dns-rr-smtp-03.txt.

[6] Spamassassin, http://spamassassin.org/.

[7] P. Graham, A Plan for Spam, http://paulgraham.com/ spam.html

[8] P. Graham, Better Bayesian Filtering, *Spam conference*, Boston, USA, 2003, http://spamconference.org/proceedings2003.html.

[9] E. Gabber, M. Jakobsson, Y. Matias, A. Mayer, Curbing Junk Email via secure Classification, *Financial Cryptography '98*, Anguilla, British West Indies, 1998, 198–213.

[10] R. J. Hall, Channels: Avoiding unwanted electronic mail, *the 1996 DIMACS Symposium on Network Threats*, Piscataway, USA, 1996, 85–103.

[11] K. Nabeya, Bsfilter, http://www.h2.dion.ne.jp/˜nabeken/bsfilter/

[12] Bogofilter, http://bogofilter.sourceforge.net/.

[13] Popfile, http://popfile.sourceforge.net/.

[14] Ruby, http://www.ruby-lang.org/en/.