# A New Approach for Detecting SMTPFA Based on Entropy Measurement

Hsing-Chung Chen, Jai-Zong Sun, Shian-Shyong Tseng, Chien-Erh Weng



# A New Approach for Detecting SMTPFA Based on Entropy Measurement

Hsing-Chung Chen[1,*], Jai-Zong Sun[2], Shian-Shyong Tseng[1], Chien-Erh Weng[3]

[1]Department of Computer Science and Information Engineering, Asia University, Taichung County, Taiwan 41354
[2]Institute of Computer Science and Information Engineering, Asia University, Taichung County, Taiwan 41354
[3]Department of Electronic Communication Engineering, National Kaohsiung Marine University Kaohsiung, Taiwan
cdma2000@asia.edu.tw, asiaphd10026@gmail.com,
sstseng@asia.edu.tw, ceweng@mail.nkmu.edu.tw

**Abstract.** In this paper, we propose a new approach of detecting a kind of Simple Mail Transfer Protocol Flooding Attack (SMTPFA for short) based on entropy measurement. We will calculate the entropy values from the received packets flow. Further checking its entropy value compared with the values of abnormal entropy, we then use it to detect this server whether is suffered some attacks from hacker. The scheme can easily detect SMTPFA, and monitor the real-time status of SMTP server.

**Keywords:** SMTP, Entropy, Attack detecting, SMTP flooding attack

## 1 Introduction

In recent years, with the rapid development of the networks, people communicate of long range gradually shift from use the traditional post letter becoming to use e-mail delivery. Today, e-mail has become one of necessary communication tools for Internet users. In 1982, the early stage of e-mail development, the SMTP (Simple Mail Transfer Protocol) is formulated in RFC 821[5]. Owing to RFC 1939 [7], RFC 2821 [6] and RFC 3461 [8] are formulated in the RFC standard, the e-mail protocol has been gradually completed.

A simple e-mail server (SMTP server, for short hereinafter) has a lot of users, so it became an important attacked target in the network. The ways of attacks includes SMTP Flooding Attack (SMTPFA), spam attacks and the malicious attachment etc. in e-mail [1, 2, 10, 12]. The SMTPFA will increase the loading of the server. In this paper, we propose a new approach for detecting SMTPFA based on entropy operation. Then, we use the entropy operation to analyse the received packets, in order to distinguish normal packets and abnormal packets from SMTP message flow, and then calculate the corresponding information parameters. Therefore, the information parameter will be

used to describe the status of the serving server. According to the value of this status, the server will determine whether it is suffered by SMTPFA.

The remainder of this paper is organized as follows. Section 2 describes the SMTP and entropy operation related work. In the Section 3, we propose a new approach for detecting SMTPFA based on entropy operation, and describe how to calculate the parameters of server status. Finally, we draw conclusions in Section 4.

## 2 Related Work

In our proposed approach, we use the entropy measurement to detect the behaviour of the SMTPFA. Therefore, in Section 2, we will describe the normal message flows of SMTP standard [5, 8], and the entropy operations [2, 3].

### 2.1 SMTP

First, SMTP had been defined in the RFC 821[5]. It is an independent subsystem in special communication system. In this communication system, it only needs a reliable channel to transmit the related sequence message flows. SMTP has an important simple delivering e-mail protocol which it can forward an e-mail between two different networks. The architecture of SMTP is shown in Fig. 1. In the SMTP architecture, it consists of a Sender, a sender-SMTP, a receiver-SMTP and a Receiver. When a Sender (user or file server) will connect to another receiver, it will send a request message of Establishes Connection to the sender-SMTP. Then, the sender-SMTP will establish a two-way transmission channel in order to connect the Receiver. The receiver-SMTP will be as a destination point or a relay point. Thus, the sender-SMTP will send the related SMTP commands to the receiver-SMTP. Finally, the receiver-SMTP will follow these commands to send back a SMTP response message to sender-SMTP. According to the above steps, if the command-respond pair has been completed during one normal time-period, it means that a round of SMTP session has been completed. The established SMTP message flows are divided into seven stages [5, 8] as below: Establishes Connection, HELO, MAIL FROM, RCPT TO, DATA, DATA TRANSFER, and QUIT. The SMTP message flows are shown in Fig. 2.

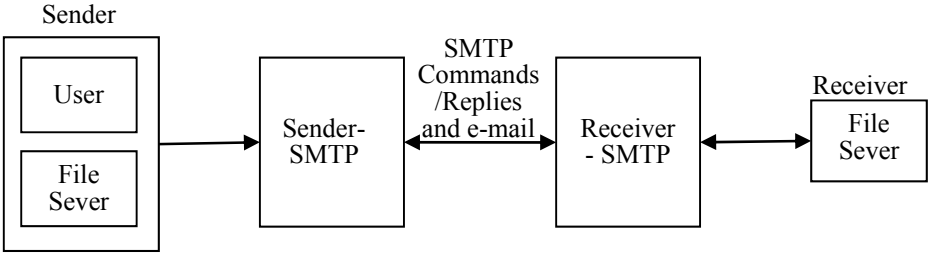



**Fig. 1.** The SMTP architecture

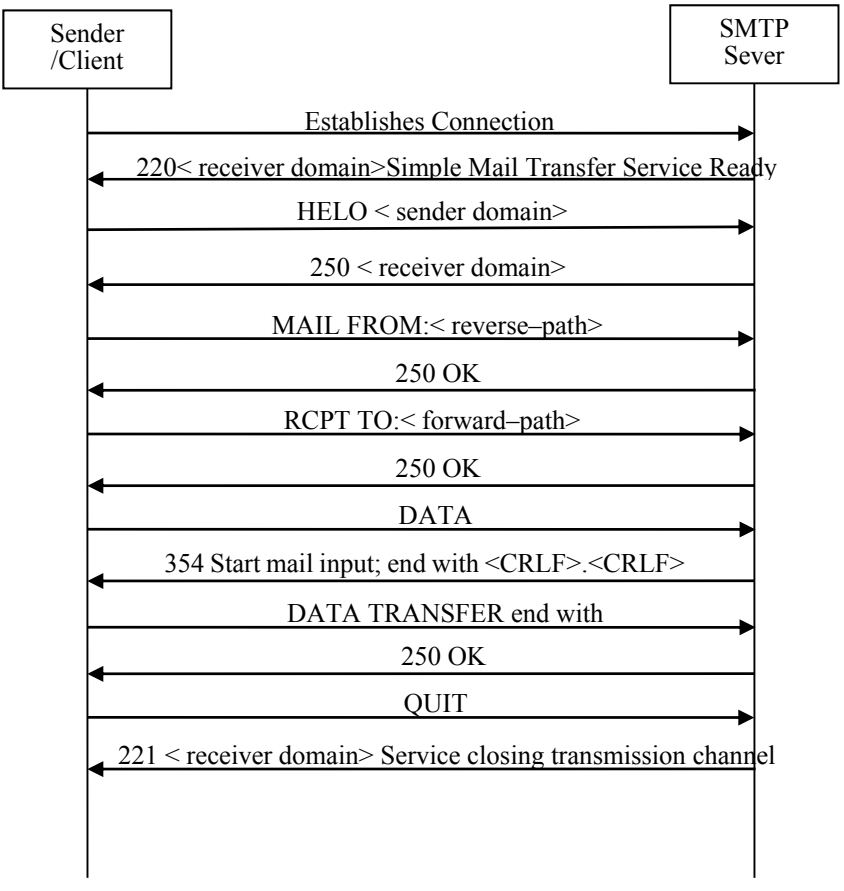



**Fig. 2.** SMTP message flows

At first, in Establishes Connection, Client (e-mail sender) will send a request of Establishes Connection to SMTP server, and prepare to forward an e-mail. After SMTP server getting the requesting connection, it will send a message '220' which means the SMTP server has prepared completely. When client gets the message "220" from SMTP server, it will send a command as "HELO < sender domain>" to SMTP server. After SMTP server gets the command as "HELO", a buffer will be initialized, and sent back the message "250 < receiver domain>" which means the command of Client is correct. Then, Client will write the source address into "MAIL FROM: < reverse–path>" command. The main purpose of this step is that when some errors have occurred during e-mail delivered path, it will return error messages of the e-mail to Client via reverse-path. If the e-mail address is inerrancy, the SMTP server will response the message "250", else it will response the message "550" which means the e-mail address doesn't exist. After Client gets the message "250", it will send back the "RCPT TO :< forward–path>" command. When the above steps have been completed, the client will

use the “DATA” command to notify the SMTP server. Then, the data will be sent via the file server. When SMTP server gets command, it will response the message “354” to notify Client “You can start to upload the e-mail, and add “<CRLF>.<CRLF>” at the end of this e-mail. After getting “<CRLF>.<CRLF>”, the SMTP server then responses the message “250” that means the e-mail has finished successfully the work. Finally, if Client doesn’t want to uses e-mail functions, she/he will send back the “QUIT” command to notify the SMTP server. Then, the SMTP server further response the message “221” which means as “shut down the connected service”.

### 2.2 Entropy Operation

In the Information Theory, Entropy is an approach used to measure the uncertainty or randomness in random variable [4, 9]. Entropy measurement approach is proposed by Shannon [3] and Weaver [11]. In the entropy operation, a random entropy value $X \in \{x_1, x_2, x_3, ..., x_n\}$, the entropy calculation formula [4, 9] as below:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log P(x_i) \tag{1}$$

where $P(x_i) = \frac{m_i}{m}$, $m = \sum_{i=1}^{n} m_i$, $m_i$ is the observation frequency or numbers of the $x_i$ from X. It can represent [4, 9] as:

$$H(X) = -\sum_{i=1}^{n} \left( \frac{m_i}{m} \right) \log P\left( \frac{m_i}{m} \right) \tag{2}$$

For example, if we throw a coin according to the formula (1) and (2), the positive and negative entropy values will be shown as follows:

$$H(positive) = -\left( 0.5 \log_2 \frac{0.5}{1} + 0.5 \log_2 \frac{0.5}{1} \right) = 1 = H(negative)$$

Throwing a coin will face the probability of 99%, the positive entropy value is as

$$H(positive) = -\left( 0.99 \log_2 \frac{0.99}{1} \right) \cong 0.014.$$

And the negative entropy value is as

$$H(negative) = -\left( 0.01 \log_2 \frac{0.01}{1} \right) \cong 0.066.$$

From the example above, we know that the coin is thrown according to the positive and negative probability to determine the entropy. The entropy value is inversely proportional to the probability value. With this feature, the value of results we calculate has dependability.

## 3 A New Approach for Detecting SMTPFA

When SMTP server is under the SMTPFA, it will have a large number of request packets into SMTP server. In SMTPFA, the request packets will sent to SMTP server from clients, and the connection will be established with the SMTP server. But clients will not be sent the command packets again. At the same time, there have request packets from another client into SMTP server. Therefore, the server resources will be reduced ceaselessly, further increasing the loading of the server. In this session, we propose a new approach for detecting SMTPFA based on entropy operation. This approach can quickly to detect SMTPFA. In order to detect SMTPFA, we mark a server command packet and a client message packet become to a pairs. A normal packets pair (hereinafter referred to as the NPP) include one command packet and one message packet; an abnormal packets pair (hereinafter referred to as the APP) just include one message packet and no command packet. Then, we calculate the normal and abnormal packets entropy value in the SMTP message flow. By using this entropy to determine whether the server was affected by SMTPFA.

In general, when SMTPFA has starting, the device other than the server may not work. However, in our proposed method, we assume that the SMTP server crashing by SMTPFA. This represents the SMTPFA will attack the server directly, and does not affect the router and bandwidth. We will describe the process of SMTP message flow matching as below. Then, we will explain how to use the entropy operations to calculate the SMTP server status value. Finally, we describe a server status change of entropy situation when a server is under SMTPFA. Some notations used in the paper are given in Table 1.

**Table 1.** Notations

| Notations | Definitions | Notations | Definitions |
|---|---|---|---|
| $Port_i$ | The i-th port number | $D_{loss}$ | The fail packets are sent from client to SMTP server. |
| $S_{port_i}$ | The i-th message flow packet pair, for the port number $port_i$, which is send from SMTP server. It includes two packets: $S_{S\_port_i}$ and $S_{D\_port_i}$. | $T_w$ | The w-th unit of time slide window, w=1, 2, 3, …, n, |
| $S_{D\_port_i}$ | The i-th packet is delivered from client to SMTP server, which is one of the packet pair $S_{port_i}$. | $P$ | The total SMTP message flow pair numbers, where $P = P_n + P_a$. |
| $S_{S\_port_i}$ | The i-th packet is delivered from SMTP server to client, which is one of the packet pair $S_{port_i}$. | $P_n$ | The number of normal SMTP message flow pair, where $P_n = \{P_{n_1}, P_{n_2}, P_{n_3}, \ldots, P_{n_j}\}$, j=1, 2, 3, …, n |

| | | | |
|---|---|---|---|
| $D_{port_i}$ | The i-th message flow pair, for the port number $port_i$, which is delivered from client to SMTP server. It includes both packets: $D_{S\_port_i}$ and $D_{D\_port_i}$. | $P_a$ | The number of abnormal SMTP message flow pair, where $P_a = \{P_{a_1}, P_{a_2}, P_{a_3}, \ldots, P_{a_j}\}$, j=1, 2, 3, …, n。 |
| $D_{S\_port_i}$ | The i-th packet is delivered from SMTP server to client, which is one of the packet pair $D_{D\_port_i}$. | $H(X)$ | An entropy value set. $X = \{x_1, x_2, x_3, \ldots, x_j\}$, j=1, 2, 3, …, n。 |
| $D_{D\_port_i}$ | The i-th packet is delivered from SMTP server to client, which is one of the packet pair $D_{port_i}$. | $S$ | The status value of SMTP Server |

### 3.1 SMTP Message Flow

*Definition* 1. A completion SMTP message flow contains six rounds, which not include the steps of Establishes Connection and Service closing transmission channel. The round 0 is represented to the packet pair of establishment connection, and the round 7 is represented the packet pair of closing transmission channel. There is only one packet in the round 0 and round 7. After the SMTP connection being established, the message flow will be divided into a pair of two packets matching. The packet pair is $(S_{port_i}, D_{port_i})$, where $(S_{port_i}, D_{port_i}) = ((S_{S\_port_i}, S_{D\_port_i}), (D_{D\_port_i}, D_{S\_port_i}))$. It is recorded by the SMTP server.

According to Definition 1, In SMTP message flow; there have two packets in one pair. Every packets pair includes a source and destination port number, and it is used to identify the sender and the receiver. For example, in the round 1, SMTP server will send a message "220 : Service already" to client. The packet pair is called $(S_{S\_port_i}, S_{D\_port_i})$, it means the packet that send to Client form SMTP server. When the destination of client gets the packet, it will send a HELO packet to SMTP server, it is called $(D_{D\_port_i}, D_{S\_port_i})$. This packet means that send to SMTP server form Client. Therefore, this transmission process is called a successful packet pair $(S_{port_i}, D_{port_i})$. If there has a packet is transmission fail or over the SMTP server waiting time from client, it's called a fail packet pair $(S_{port_i}, D_{loss})$, and record into the SMTP server. SMTP packets pair message flow as shown Fig. 3.

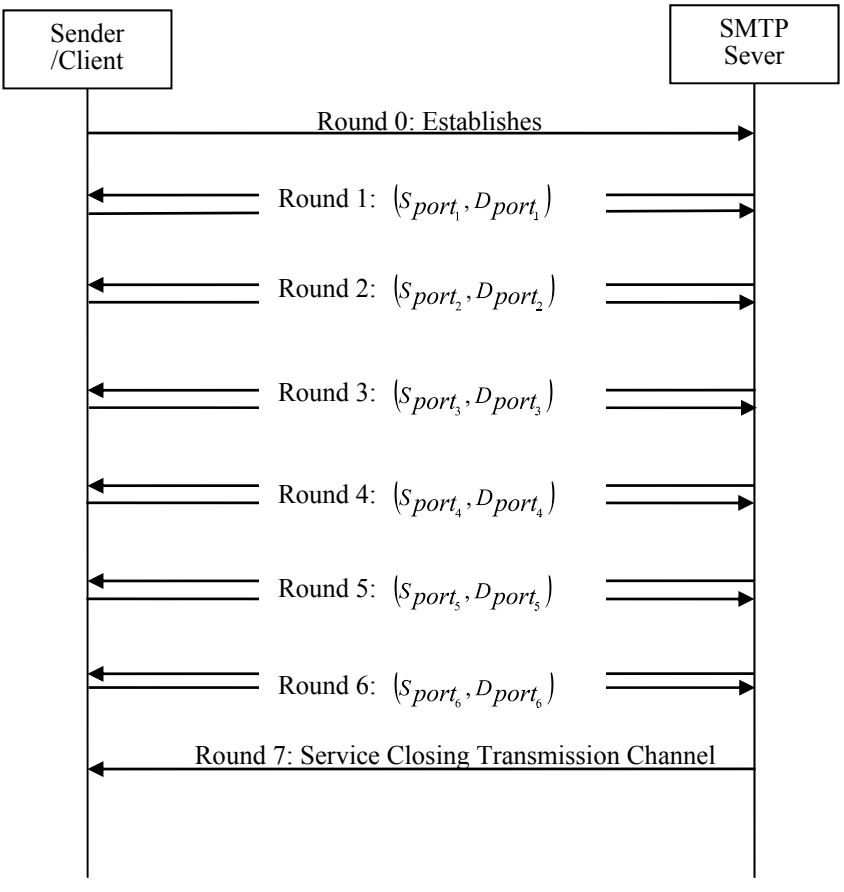



**Fig. 3.** SMTP packet pair message flows.

### 3.2 A approach for detecting SMTPFA using entropy

*Definition 2. In the SMTP, it has a NPP at least, to represent the SMTP server and the client has completed at least the starting of SMTP connection steps.*

In the approach of detecting the SMTPFA, SMTP message flows are divided into two one-group pairing according to the description of Section 3.1. Then, we use the entropy operation to calculate the SMTP packets pair entropy values for the normal message flows and abnormal message flows. The entropy formulas are listed as follows.

$$Entropy: H(x_1) = -\left(\sum \frac{P_n}{P} \log_2 \frac{P_n}{P}\right), \tag{3}$$

$$Entropy: H(x_2) = -\left(\sum \frac{P_a}{P} \log_2 \frac{P_a}{P}\right), \tag{4}$$

where $P=P_n+P_a$.

By formula (3), we calculate the normal packets pair entropy $H(x_1)$ at $T_w$. If the normal number of packets pair is more, its mean the normal packets pair entropy $P_n$ will less. Otherwise, the number of normal packets pair is less, its mean the normal packets pair entropy $P_n$ will more. In formula (4), we calculate the normal packets pair entropy $H(x_2)$ at $T_w$. If the number of abnormal packets pair is more, its mean the abnormal packets pair entropy $P_a$ will less. Otherwise the number of abnormal packets pair is less; it means the abnormal packets pair entropy $P_n$ will more. According to Definition 2, if all of the packets pair is abnormal in SMTP message flow, it means all of the packets are lost, and the SMTP server and Client connections will not start. Therefore, in the SMTP message flow must be at least a normal packets pair to represent server and client is established a connection, and complete the SMTP steps of establish connection. Finally, according to formula (1) and formula (2), we get the intersection point of the entropy of the normal packets pair entropy and abnormal packets pair, as shown in Fig.4. When the server status values are more near to the intersection point, it indicating the server's security will be lower. The following we use the algorithm to describe the server status value.

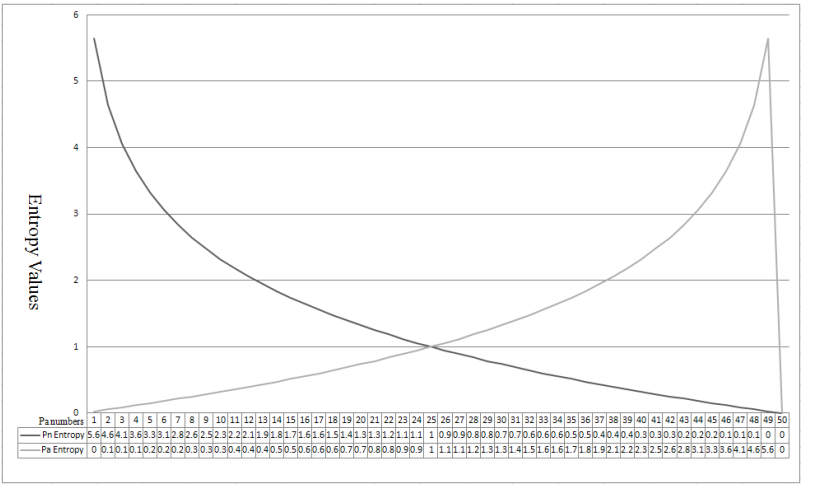



**Fig. 4.** The intersection point of entropy.

***Algorithm***

**Input:** $\left(H(P_n), H(P_a)\right)$, a SMTP message flow pair is including two Entropy value for normal packets and abnormal packets in $T_w$.

**Output:** Result values S of The SMTP server status in $T_w$. The value is 0, 1 or -1, where its safety, threshold limit value, dangerous, respectively.

**Begin**

$$H(P_n) \leftarrow -\left(\sum \frac{P_n}{P} \log 2 \frac{P_n}{P}\right)$$

$$H(P_a) \leftarrow -\left(\sum \frac{P_a}{P} \log 2 \frac{P_a}{P}\right)$$

**for** $P_n = P$ to $P - P_n$ do

**for** $P_a = P$ to $P - P_a$ do

Generate $(H(P_n), H(P_a))$;

**if** $H(P_n) - H(P_a) < 0$ **then**

return -1;

**else if** $H(P_n) - H(P_a) \cong 0$ **then**

return 0;

**else** $H(P_n) - H(P_a) > 0$ **then**

return 1;

end

end

end

return S;

end

□

Both NPP entropy value and APP entropy value are same almost, then $H(P_n) - H(P_a) \cong 0$, and return 0 mean the status on behalf of the server in the security value of the critical point. When the NPP entropy value adds the APP entropy value will less than 0, $H(P_n) - H(P_a) < 0$, its mean the SMTP server is under the danger status. If the server in this case, it will be attacked by SMTPFA or the SMTP server will overload. When the NPP entropy value adds the APP entropy value will greater than 0, $H(P_n) - H(P_a) > 0$, its mean the SMTP server is safe status. Finally, return the server status value $S$, and according to the value determine whether the SMTP server was affected by SMTPFA.

### 3.3 An example for detecting of SMTPFA

In the time slide window $T_W = \{T_1, T_2, T_3, ..., T_{10}\}$, the SMTP packets pair is 100 in a time slide window, and the packets total is 1000, where:

$$P_n = \{P_{n_1}, P_{n_2}, P_{n_3}, ..., P_{n_{10}}\} = \{100, 95, 87, 85, 76, 67, 51, 32, 22, 19\}$$

$P_{n_1}$ entropy: $H(P_{n_1}) = -\left(\sum \frac{P_{n_1}}{P} \log_2 \frac{P_{n_1}}{P}\right) = -\frac{100}{100} \log \frac{100}{100} = 0$

$P_{n_2}$ entropy: $H(P_{n_2}) = -\left(\sum \frac{P_{n_2}}{P} \log_2 \frac{P_{n_2}}{P}\right) = -\frac{95}{100} \log \frac{95}{100} = 0.074$

$\vdots$

$P_{n_{10}}$ entropy: $H(P_{n_{10}}) = -\left(\sum \frac{P_{n_{10}}}{P} \log_2 \frac{P_{n_{10}}}{P}\right) = -\frac{19}{100} \log \frac{19}{100} = 2.396$

$H(P_n) = \{0, 0.074, 0.201, 0.234, 0.396, 0.578, 0.972, 1.644, 2.185, 2.396\}$ Then,

$P_a = \{P_{a_1}, P_{a_2}, P_{a_3}, ..., P_{a_{10}}\} = \{0, 5, 13, 15, 24, 33, 49, 68, 78, 81\}$

$P_{a_1}$ entropy:

$H(P_{a_1}) = -\left(\sum \frac{P_{a_1}}{P} \log_2 \frac{P_{a_1}}{P}\right) = -\frac{0}{100} \log \frac{0}{100} = 0$

$H(P_{a_2}) = -\left(\sum \frac{P_{a_2}}{P} \log_2 \frac{P_{a_2}}{P}\right) = -\frac{5}{100} \log \frac{5}{100} = 4.322$

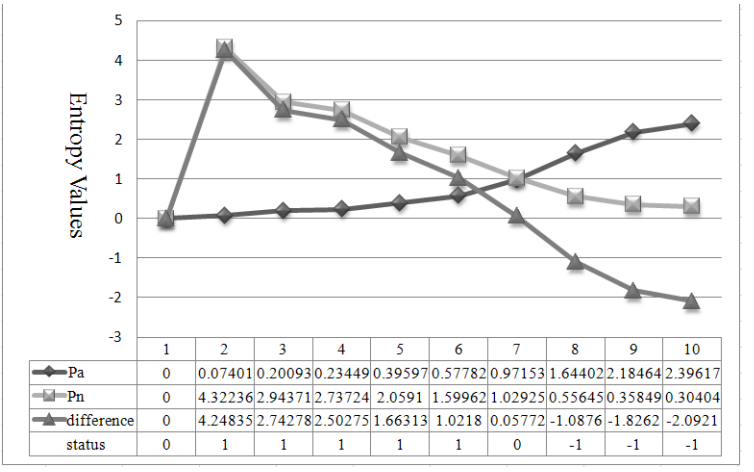
$P_{a_2}$ entropy:

$\vdots$

$P_{a_{10}}$ entropy: $H(P_{a_{10}}) = -\left(\sum \frac{P_{a_{10}}}{P} \log_2 \frac{P_{a_{10}}}{P}\right) = -\frac{81}{100} \log \frac{81}{100} = 0.304$

$H(P_a) = \{0, 4.322, 2.944, 2.737, 2.059, 1.6, 1.029, 0.556, 0.358, 0.304\}$.

In the above calculation process, $H(P_n)$ is NPP entropy, $H(P_a)$ is APP entropy. We prove that the description of Section 3.2 is correct: the smaller the entropy value, the number of normal packets pair will be greater; the smaller the value, the fewer the number representing the abnormal packets. Then, we will add $H(P_a)$ and $H(P_n)$ in the same time slide window $T_w$, and calculate the status value to behalf the SMTP server status. Finally, the SMTP server status, $H(P_a)$ and $H(P_n)$ are show in Fig. 5.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Pa | 0 | 0.07401 | 0.20093 | 0.23449 | 0.39597 | 0.57782 | 0.97153 | 1.64402 | 2.18464 | 2.39617 |
| Pn | 0 | 4.32236 | 2.94371 | 2.73724 | 2.0591 | 1.59962 | 1.02925 | 0.55645 | 0.35849 | 0.30404 |
| difference | 0 | 4.24835 | 2.74278 | 2.50275 | 1.66313 | 1.0218 | 0.05772 | -1.0876 | -1.8262 | -2.0921 |
| status | 0 | 1 | 1 | 1 | 1 | 1 | 0 | -1 | -1 | -1 |

**Fig. 5.** The status of SMTP server, $H(P_a)$ and $H(P_n)$

## 4 Conclusions

In this paper, we propose a new approach for detecting SMTPFA based on entropy measurement. By using this method, we can quickly analyse the current status of the SMTP server, and determine whether the server is attacked by SMTPFA or not. This approach can not only be applied to different SMTP servers, but also monitor the real-time status of SMTP server. Finally, according to the status value of STMP server by using the entropy measurement we proposed, it can detect SMTPFA easily and quickly.

## Acknowledgements

This work was supported in part by Asia University, Taiwan, under Grant 100-asia-34, also by the National Science Council, Taiwan, Republic of China, under Grant NSC99-2221-E-468-011.

## References

1. A. J. O'Donnell, "The Evolutionary Microcosm of Stock Spam," Security & Privacy, IEEE, pp. 70-75, 2007.
2. Bass, T. Watt, G., A simple framework for filtering queued SMTP e-mail, MILCOM 97 Proceedings, vol. 3, Nov 1997, Pages: 1140 - 1144.
3. C. E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, 1948.
4. "Information Entropy," Available from: http://www.absoluteastronomy.com/topics/Information_entropy.
5. J. B. Postel, "A SIMPLE MAIL TRANSFER PROTOCOL," RFC821, 1982.

6. J. Klensin, “SIMPLE MAIL TRANSPORT PROTOCOL,” RFC2821, 2001.
7. J. Myers, M. Rose, “Post Office Protocol - Version 3,” RFC 1939, 1996.
8. K. Moore, “Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs),” RFC 3461, 2003.
9. S. Russell, P. Norvig, Artificial Intelligence- A Modern Approach 3/E, 2011.
10. T. Bass, A. Freyre, D. Gruber and G. Watt, “E-Mail Bombs and Countermeasure: Cyber Attack on Availability and Brand Integrity,” IEEE Network, Vol. 12, No. 2, p10-17, 1998.
11. W. Weaver and C. E. Shannon, The Mathematical Theory of Communication, 1949, republished in paperback 1963.
12. X. Wang, S. Chellappan, P. Boyer, D. Xuan, “On the effectiveness of secure overlay forwarding systems under intelligent distributed DoS attacks,” IEEE Transactions on Parallel and Distributed Systems, pp.619-632, 2006.