British Journal of Mathematics & Computer Science
12(5): 1-11, 2016, Article no.BJMCS.21889
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

SCIENCEDOMAIN

Perfect-Mail: A Secure E-mail Protocol with Perfect Forward Secrecy

Baoyuan Kang[1*] and Danhui Xu[1]

[1]*School of Computer Science and Software, Tianjin Polytechnic University, Tianjin, 300387, China.*

Article Information

DOI: 10.9734/BJMCS/2016/21889
Editor(s):
(1) Kai-Long Hsiao, Taiwan Shoufu University, Taiwan.
Reviewers:
(1) Anand Nayyar, KCL Institute of Management and Technology, Jalandhar, Punjab, India.
(2) Valasani Usha Shree, J.B.Institute of Engineering and Technology, India.
(3) S. Akhila, BMS College of Engineering, Bangalore, India.
Complete Peer review History: http://sciencedomain.org/review-history/12100

Original Research Article

Received: 08 September 2015
Accepted: 08 October 2015
Published: 06 November 2015

Abstract

With the rapid development of Internet, e-mail has become an essential communication tool. But, the security of e-mail communications is an important issue. Recently, Chen et al. [6] proposed a new protocol of wide use for e-mail. Chen et al. claimed that the proposed protocol is skillfully designed to achieve perfect forward secrecy and end to end security as well as to satisfy the requirements of confidentiality, origin, integrity and easy key management. But, in this paper, we show that Chen et al.'s protocol suffers from the e-mail server impersonation attack, mail content confidentiality attack and replay attack. Moreover, we give an improvement on Chen et al.'s protocol to overcome its security weaknesses, and propose the perfect-mail, a secure e-mail protocol with perfect forward secrecy. It is concluded by analysis that the improved protocol provides the perfect forward secrecy and resists replay attack, impersonation attack, and mail content confidentiality attack. But the communication cost of improved protocol is equal to that of Chen et al.'s protocol, and the computing cost of improved protocol is only added by two signature verification.

Keywords: Cryptography; secure protocol; E-mail protocol; security.

1 Introduction

Electronic mail, e-mail in short, has been widely used instead of traditional communication established by pen and paper. Moreover, with the rapid development of Internet, e-mail has become an essential communication tool. Modern e-mail system transfer not only text but also electronic documents, voice, and financial transactions. So, the security of e-mail communications is an important issue. Unfortunately the

**Corresponding author: E-mail: baoyuankang@aliyun.com, mixueren123123@sina.com;*

basic e-mail protocol does not provide the confidentiality and integrity service. Bacard [1] introduced some security requirements in e-mail systems. Since then, several security protocols such as, PGP [2], PEM [3] and S/MIME [4] have been designed to provide confidentiality and authentication of e-mail system.

However, these protocols cannot provide perfect forward secrecy [5] because once the secret key of the receiver is disclosed, all previous used short-term keys will also be opened and hence previous e-mail will be learned.

It is noted that early e-mail protocols take only a single e-mail server into account. But, in practice, it is common that the e-mail sender and receiver any register at different e-mail servers. Recently, Chen et al. [6] took into account the scenario that the e-mail sender and the recipient register at different servers and proposed a new protocol of wide use for e-mail. Chen et al. claimed that the proposed protocol is skillfully designed to achieve perfect forward secrecy and end to end security as well as to satisfy the requirements of confidentiality, origin, integrity and easy key management. But, in this paper, we show that Chen et al.'s protocol suffers from the e-mail server impersonation attack, mail content confidentiality attack and replay attack. Moreover, we give an improvement on Chen et al.'s protocol, and propose the perfect-mail, a secure e-mail protocol with perfect forward secrecy. We also discuss the security of the improved protocol. The improved protocol provides the perfect forward secrecy and resists replay attack, impersonation attack, and mail content confidentiality attack.

This article is organized as follows. Section 2 discusses the related work. We review Chen et al.'s protocol in Section 3 and point out its flaws in Section 4. In Section 5, we give an improvement on Chen et al.'s protocol. The security analysis of the improved protocol, the perfect-mail, is discussed in Section 6. Finally, conclusions are given in Section 7.

2 Related Works

In order to provide perfect forward secrecy, Sun et al. [5] proposed two new e-mail protocols. However, in 2006, Dent [7] pointed out Sun et al.'s protocols do not provide perfect forward secrecy as claimed. Later, Kim et al. [8] proposed an improved version of Sun et al.'s protocols to overcome this weakness. But, in 2010, Chang et al. [9] showed that Kim et al.'s protocols suffer from the well-known man-in-the-middle attack and consequently do not achieve perfect forward secrecy. In 2007，Kwon et al. [10] proposed a password-based e-mail protocol for mobile devices. However too many modular exponentiation operations in their protocol might cause mobile devices consume battery power expeditiously [9]. In 2011, Chang et al. [11] pointed out some drawbacks of existing e-mail protocol and proposed an efficient e-mail protocol for mobile devices. In 2012, Wong et al. [12] proposed a secure e-mail protocol with perfect forward secrecy.

Certified e-mail protocol is a fair exchange of a message for receipt between two potentially mistrusting parties over the network. In 2013, Gao et al. [13] proposed an improved certified e-mail protocol meeting confidentiality and non-repudiation. In 2013, Wang et al. [14] developed a novel certified e-mail protocol in id-based setting that employed an off-line semi-trusted third party STTP for wireless networks. In 2014, Draper-Gil et al. [15] proposed an optimistic certified e-mail protocol for the current Internet e-mail architecture.

3 Review of Chen et al.'s E-mail Protocol

In this section, we review Chen et al.'s e-mail protocol [6]. Chen et al.'s protocol consists of three phase: registration, sending, and receiving.

3.1 Registration

Either the sender or the recipient has to register at an individual e-mail server at the beginning. For example, when a participant A (resp. B) registers at e-mail server S_A (resp. S_B), it implies that A shares password

Q_1 with S_A. A submits ID_A and $g^{aQ_1} \bmod n$ to S_A where n is a big prime number, g is a generator with order $n-1$ over $GF(n)$, and a is a random number. S_A computes the registration information ($g^a \bmod n$) with Q_1^{-1} and stores ($g^a \bmod n$). Likewise, the participant B shares Q_2 with e-mail server S_B. S_B stores ($g^b \bmod n$) for B. The e-mail server S_A and S_B also share a password K, MAC denotes a message authentication code. $[\cdot]_K$ denotes the symmetric encryption with the key K. For simplicity, $'\bmod n'$ is omitted hereafter.

3.2 Sending Phase

When sender A intends to send an e-mail to recipient B, the operation goes as follows:

Step 1: $A \to S_A$: Request.

If A wants to deliver an e-mail to B, he should send the request to S_A firstly.

Step 2: $S_A \to S_B$: Request.

S_A forwards the request to S_B to ask for the registration information of B

Step 3: $S_B \to S_A$: $ID_B, g^b, MAC_K(ID_B, g^b)$

S_B finds the registration information g^b of B. Then S_B computes the MAC value of ID_B, g^b with K, and sends $ID_B, g^b, MAC_K(ID_B, g^b)$ to S_A.

Step 4: $S_A \to A$: $ID_B, g^b, MAC_{Q_1}(ID_B, g^b)$

In order to check the validation of the received message, S_A computes $MAC_K(ID_B, g^b)$ and checks if the computed MAC value is equal to the received MAC value. If it holds, S_A computes the MAC value of ID_B, g^b with Q_1 and sends $ID_B, g^b, MAC_{Q_1}(ID_B, g^b)$ to A.

Step 5: $A \to S_A: ID_A, ID_B, [M]_{g^{xb}}, g^x, MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x)$.

Upon receiving the message, A computes $MAC_{Q_1}(ID_B, g^b)$ and checks if the computed MAC value is equal to the received MAC value. If it holds, A computes g^x with a random number x and g^{xb} by computing $(g^b)^x$. A encrypts mail content M with g^{xb}. Then A computes the MAC value of $ID_A, ID_B, [M]_{(g^x b)}, g^x$ with Q_1 and sends

$$ID_A, ID_B, [M]_{(g^x b)}, g^x, MAC_{(Q_1)}(ID_A, ID_B, [M]_{(g^x b)}, g^x)$$

to S_A.

Step 6: $S_A \rightarrow S_B : ID_A, ID_B, [M]_{(g^{xb})}, g^x, MAC_K(ID_A, ID_B, [M]_{(g^{xb})}, g^x)$.

S_A checks the validation of the received message. he computes $MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x)$ and checks if the computed MAC value is equal to the received MAC value. If it holds, S_A computes the MAC value of $ID_A, ID_B, [M]_{(g^x b)}, g^x$ with K and sends

$$ID_A, ID_B, [M]_{(g^{xb})}, g^x, MAC_K(ID_A, ID_B, [M]_{(g^{xb})}, g^x)$$

to S_B. After receiving the message, S_B stores the e-mail message for B.

3.3 Receiving Phase

Step 7: $B \rightarrow S_B$: $ID_B, \mathrm{g}^{b'}, MAC_{Q_2}(ID_B, \mathrm{g}^{b'}, \mathrm{g}^b)$.

When B is on-line and intends to check e-mails, he will compute $\mathrm{g}^{b'}$ with a new random number b' and $MAC_{Q_2}(ID_B, \mathrm{g}^{b'}, \mathrm{g}^b)$. Then B sends

$$ID_B, \mathrm{g}^{b'Q_2}, MAC_{Q_2}(ID_B, \mathrm{g}^{b'}, \mathrm{g}^b)$$

to S_B

Step 8: $S_B \rightarrow B$: $ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, MAC_{Q_2}(ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^{b'}, g^b)$

Upon S_B receiving the message, S_B verifies $MAC_{Q_2}(ID_B, \mathrm{g}^{b'}, \mathrm{g}^b)$. If the verification fails, S_B will reject the request from B. Otherwise, S_B update g^b with $\mathrm{g}^{b'}$. Lastly, S_B computes the MAC value of $ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, g^{b'}$ with Q_2 and sends

$$ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, MAC_{Q_2}(ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, g^{b'})$$

to B.

When B receives the message from S_B, he computes

$$MAC_{Q_2}(ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, g^{b'}).$$

B checks if the computed MAC value. If it holds, he computes g^{xb} by computing $(g^x)^b$ to decrypt $[\mathrm{M}]_{g^{xb}}$.

4 The Cryptanalysis of Chen et al.'s Protocol

In this section, we show that Chen et al.'s protocol suffers from the e-mail server impersonation attack, mail content confidentiality attack and replay attack.

4.1 The E-mail Server Impersonation Attack

In Chen et al.'s protocol, the e-mail server S_B can impersonate the e-mail sender A to send message to B.

In fact, when S_B receivers $g^{b'}$ in step 7, S_B can pick a random number x' and computes $g^{x'}$. Then S_B computes

$$[M']_{g^{x'b}}\ , MAC_{Q_2}(ID_A, ID_B, [M']_{g^{x'b}}, g^{b'}, g^b).$$

Where M' is the mail content that S_B wants to impersonate the e-mail sender A to send to B. Then S_B sends

$$ID_A, ID_B, [M']_{g^{x'b}}, g^{x'}, MAC_{Q_2}(ID_A, ID_B, [M']_{g^{x'b}}, g^{b'}, g^b)$$

to B. Receiving the message, B cannot find any problem by checking the MAC value and believe M' is the mail content which the sender A want to send him. So, the e-mail server S_B successfully impersonate the sender A to send message to the receiver B.

4.2 Replay Attack

In Chen et al.'s protocol, when an attacker intercepts the message $ID_A, ID_B, [M]_{g^{xb}}, g^x, MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x)$ in step 5, he can use it in future to implement replay attack. In next procedure of A sending e-mail to B, the attacker can send the intercepted message

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x)$$

to S_A in step 5. S_A cannot find any problem. Then S_A sends

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, MAC_K(ID_A, ID_B, [M]_{g^{xb}}, g^x)$$

to S_B. In step 6, S_B also cannot find any problem. Then S_B sends

$$ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^x, MAC_{Q_2}(ID_A, ID_B, [\mathrm{M}]_{g^{xb}}, g^{b'}, g^b)$$

to B. In step 8, the message also satisfies the verification. So, the attacker successfully implements replay attack. Of course, at the end of the replay attack, the mail content got by the receiver B may not be M, because the personal information g^b might have replaced by $g^{b'}$.

4.3 Mail Content Confidentiality Attack

In step 4 of Chen et al.'s protocol, the mail server S_A can pick a random number c and send $ID_B, g^c, MAC_{Q_1}(ID_B, g^c)$ to the e-mail sender A. Then in step 5 when S_A receivers the message $[M]_{g^{xc}}, g^x$, S_A can compute $g^{xc} = (g^x)^c$ and obtain the mail content by decrypting $[M]_{g^{xc}}$. Then S_A can continue performing step 6. At the end of the protocol, the receiver B may get a false mail content since $g^c \neq g^b$.

5 A Secure E-mail Protocol with Perfect Forward Secrecy

5.1 Registration

The registration phase of the improved protocol is essentially identical to that of Chen et al.'s protocol. The mail sender A shares a password Q_1 with his mail server S_A. The mail receiver B shares a password Q_2 with his mail server S_B. S_A and S_B also share a password K, MAC denotes a message authentication code. $[\cdot]_K$ denotes the symmetric encryption with the key K. But, the personal information of the e-mail sender A is g^a and $Sig_{SK_A}(g^a)$. Where SK_A is the private key of A, $Sig_{SK_A}(g^a)$ is the signature generated by A. Likewise, the personal information of the e-mail receiver B is g^b and $Sig_{SK_B}(g^b)$.

5.2 Sending Phase

When sender A intends to send an e-mail to the recipient B, the operation goes as follows:

Step 1: $A \rightarrow S_A$: Request.

If A wants to deliver an e-mail to B, he first sends the request to his mail server S_A.

Step 2: $S_A \rightarrow S_B$: Request.

S_A forwards the request to S_B , the recipient B's server , to ask for the registration information of B

Step 3: $S_B \rightarrow S_A$: $ID_B, g^b, Sig_{Sk_B}(g^b)$, $MAC_K(ID_B, g^b, Sig_{SK_B}(g^b))$

S_B finds $ID_B, g^b, Sig_{Sk_B}(g^b)$ of B. Then S_B computes the MAC value of $ID_B, g^b, Sig_{Sk_B}(g^b)$ with K, and sends

$$ID_B, g^b, Sig_{Sk_B}(g^b),\ MAC_K(ID_B, g^b, Sig_{SK_B}(g^b))$$

to S_A.

Step 4: $S_A \rightarrow A$: $ID_B, g^b, Sig_{Sk_B}(g^b)$, $MAC_{Q_1}(ID_B, g^b, Sig_{Sk_B}(g^b))$

S_A computes $MAC_K(ID_B, g^b, Sig_{Sk_B}(g^b))$ and checks if the computed MAC value is equal to the received MAC value. If it holds, S_A computes the MAC value of $MAC_{Q_1}(ID_B, g^b, Sig_{Sk_B}(g^b))$ and sends

$$ID_B, g^b, Sig_{Sk_B}(g^b),\ MAC_{Q_1}(ID_B, g^b, Sig_{Sk_B}(g^b))$$

to A.

Step 5: $A \rightarrow S_A$:

$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T)$ Upon receiving the message, A first verifies the signature $Sig_{SK_B}(g^b)$. Then A computes

$$MAC_{Q_1}(ID_B, g^b, Sig_{Sk_B}(g^b))$$

and checks if the computed MAC value is equal to the received MAC value. If the verifications hold, A computes g^x with a random number x and g^{xb} by computing $(g^b)^x$. A encrypts M with g^{xb}, where M is the content of the e-mail. Then A computes the MAC value of $ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T$ with Q_1 and sends

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T)$$

to S_A. Where T is time stamp.

Step 6: $S_A \rightarrow S_B$:

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_K(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T).$$

S_A computes $MAC_{Q_1}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T)$ and checks if the computed MAC value is equal to the received MAC value. If it holds, S_A computes the MAC value of $ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T$ with K and sends

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_K(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T)$$

to S_B. After receiving the message, S_B stores the e-mail message for B.

5.3 Receiving Phase

Step 7: $B \to S_B$: $ID_B, g^{b'}, Sig_{SK_B}(g^{b'}), MAC_{Q_2}(ID_B, g^{b'}, Sig_{SK_B}(g^{b'}), g^b)$.

When B checks e-mails, he will compute $g^{b'}$ with a new random number b' and $MAC_{Q_2}(ID_B, g^{b'}, Sig_{SK_B}(g^{b'}), g^b)$. . Then B sends

$$ID_B, g^{b'Q_2}, MAC_{Q_2}(ID_B, g^{b'}, Sig_{SK_B}(g^{b'}), g^b)$$

to S_B

Step 8: $S_B \to B$:

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_{Q_2}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), g^{b'}, g^b, T)$$

Upon S_B receiving the message, S_B first verifies the signature $Sig_{SK_B}(g^{b'})$. Then he verifies

$$MAC_{Q_2}(ID_B, g^{b'}, Sig_{SK_B}(g^{b'}), g^b).$$

If the verifications fail, S_B will reject the request from B. Otherwise, S_B update g^b with $g^{b'}$. Lastly, S_B computes the MAC value of

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T \text{ with } Q_2$$

and sends

$$ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), T, MAC_{Q_2}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), g^{b'}, g^b, T) \text{ to } B.$$

When B receives the message from S_B, he computes

$$MAC_{Q_2}(ID_A, ID_B, [M]_{g^{xb}}, g^x, Sig_{SK_A}(g^x), g^{b'}, g^b, T).$$

B checks if the computed MAC value is equal to the received MAC value. If it holds, he computes g^{xb} by computing $(g^x)^b$ to decrypt $[M]_{g^{xb}}$.

6 Security Analysis of the Improved Protocol

6.1 Perfect Forward Secrecy

In a protocol, if compromise of long-term keys does not compromise session keys, it's said that the protocol satisfies the perfect forward secrecy. In improved protocol, the session key g^{xb} is determined by the randomly selected secret numbers x and b. So, the session key g^{xb} has no relationship with the long-term SK_A or SK_B. Even if the attacker gets g^x and g^b by compromise of long-term keys SK_A and SK_B, the attacker also cannot get g^{xb} thanks to the difficulty of computing discrete logarithm. Therefore, the improved protocol satisfies the perfect forward secrecy.

6.2 Replay Attack

An attacker may intercept massage in step 3, step 4, step 5, step 6, step 7 and step 8. But in improved protocol the information g^b of receiver B is renewed when each receiving e-mail is finished. Secondly, time stamp T is involved in step 5, step 6, step 7 and step 8 to guarantee the freshness of transmitted messages. So, the intercepted messages are useless for the attacker to perform replay attacks.

6.3 Sender Impersonation Attack

If **a**n attacker wants to impersonate e-mail sender A to send a message to receiver B, he must know the password Q_1 or Q_2 and private key SK_A. Because in step 5, step 6 and step 8 g^x is signed by SK_A. Before decrypting the mail content, the e-mail receiver B first verifies the signature $Sig_{SK_A}(g^x)$ generated by e-mail sender A. The attacker do not know SK_A, then he cannot generate signature $Sig_{SK_A}(g^x)$. So, the attacker cannot success to perform sender impersonation attack. Of course, the e-mail server S_B cannot perform sender impersonation attack.

6.4 Mail Content Confidentiality Attack

Unlike Chen et al.'s protocol, the improved protocol can resist mail content confidentiality attack. Because in step 4 of improved protocol, the signature $Sig_{SK_B}(g^b)$ is needed, the mail server S_A cannot successfully change the information g^b of B. So, in step 5 of the improved protocol, S_A cannot decrypt $[M]_{g^{xb}}$. Of course, except the e-mail receiver B, no one can obtains the mail content.

7 Conclusion

In this paper, we show that Chen et al.'s e-mail protocol suffers from the e-mail server impersonation attack, mail content confidentiality attack and replay attack. Moreover, we give an improvement on Chen et al.'s e-mail protocol, and propose a secure e-mail protocol with perfect forward secrecy. We also discuss the security of the improved protocol. The improved protocol provides the perfect forward secrecy and resists replay attack, impersonation attack, and mail content confidentiality attack. The proposed secure e-mail protocol is more suitable to the e-mail system in our real life.

Acknowledgements

The authors would like to thank the anonymous referees for their valuable comments. This work is supported by the Applied Basic and Advanced Technology Research Programs of Tianjin (No. 15JCYBJC15900).

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Bacard A. Computer privacy handbook: A practical guide to e-mail encryption, data protection, and PGP privacy software. Berkeley, CA: Peachpit Press; 1995.

[2] Atkins D, Stallings W, Zimmermann P. PGP message exchange formats. Internet Draft; 1995.

[3] Balenson D. Privacy enhancement for internet electronic mail: Part III: Algorithms, modes, and identifiers. RFC 1423; 1993.

[4] Galvin J, Murphy G, Crocker S, Freed N. MIME object security services. RFC 1848; 1995.

[5] Sun H, Hsieh B, Hwang H. Secure e-mail protocols providing perfect forward secrecy. IEEE Communications Letters. 2005;15(8):58–60.

[6] Chen T, Wu Y. A new protocol of wide use for e-mail with perfect forward secrecy. Journal of Zhenjiang University-Science C (Computer & Electronics). 2010;11(1):74-78.

[7] Dent AW. Flaws in an e-mail protocol of Sun, Hsieh, and Hwang. IEEE Communications Letters. 2005;9(8):718–719.

[8] Kim B, Koo J, Lee D. Robust e-mail protocols with perfect forward secrecy. IEEE Communications Letters. 2006;10(6):510–512.

[9] Chang C, Lee C, Chiu Y. An efficient e-mail protocol providing perfect forward secrecy for mobile devices. International Journal of Communication Systems. 2010;23:1463–1473.

[10] Kwon JO, Jeong IR, Sakurai K, Lee DH. An efficient password-based e-mail protocol for encrypted e-mail transmissions on mobile equipment. Proceedings of the 2007 IEEE International Conference on Consumer electronics (ICCE 2007), Las Vegas, U.S.A. 2007;2–22:1–2.

[11] Chang Chin-Chen, Lee Chia-Yin. A secure e-mail protocol for mobile devices. International Journal of Innovative Computing, Information and Control. 2011;7(9):5353-5362.

[12] Wong Duncan S, Tian Xiaojian. E-mail protocols with perfect forward secrecy. International Journal of Security and Networks. 2012;7(1):1-5.

[13] Gao Yue-Xiang, Peng Dai-Yuan, Yan Li-Li. Analysis and improvement of a certified e-mail protocol**.** Journal of the University of Electronic Science and Technology of China. 2013;42(2):300-305.

[14] Wang Caifen, Lan Caihui, Niu Shufen, Cao Xiaojun, Gong Minan. An ID-based certified e-mail protocol with STTP suitable for wireless mobile environments. Journal of Computers (Finland). 2013; 8(1):3-9.

[15] Draper-Gil Gerard, Tauber Arne. An optimistic certified e-mail protocol for the current Internet e-mail architecture. 2014 IEEE Conference on Communications and Network Security, CNS. 2014; 382-390.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/12100