Enhancing Certified Email Service for Message Confidentiality

Kyi Kyi Maw, Ei Ei Khin
University of Technology (Yatanarpon Cyber City)
kyimaw83@gmail.com, ei2khin@gmail.com

Abstract

Certified email protocols to facilitate secure electronic mail delivery are necessary if the Internet is to achieve its true potential as a communications tool. So, the proposed system is aimed to provide a certified email protocol which guarantees the fairness and confidentiality of the message so that the participants can communicate via email in a fair and secret form and no one else but only the intended user can see the mail content. In order to resolve the disputes and provide fairness, off-line (optimistic) trusted third party (TTP) will be participated in this system. The system is also aimed not to reveal any secret things: the mail content and the key, to TTP so that the importance of its role can be reduced as much as possible. The key security properties of the proposed system are verified using model checking tool, Automated Validation of Internet Security Protocols and Applications (AVISPA).

1. Introduction

Electronic mail has become an essential communication tool for business. The ease of communicating over e-mail, as opposed to physical mail, makes it the communications form of choice for many people of today world. In order for e-mail to be used for important communications, some notion of certified delivery must be provided for users. Just as in the physical world, conventional mail is sufficient for most communications, but some important communications needs to be sent via certified mail.

The usage of email for official events creates some problems because, in its simplest from, the email service does not have many desirable features. Physical mail offers services such as the sending and delivery receipts, which can be used to prove the message origin and destination. Therefore, certified email tries to deal with these problems using certified email protocols. Indeed, the purpose of these protocols is to provide a procedure for the secure exchange of messages, which is resistant to possible attempts of cheating by the participants.

Therefore, certified email protocols have to guarantee several standard properties like non-repudiation of origin, non-repudiation of receipt, confidentiality and fairness. Non-repudiation means if an item has been sent from Alice to Bob, Alice cannot deny the sending of the item and also Bob cannot deny the receipt of the item. This property can be assured with the aids of non-repudiable evidence of origin (EOO) and non-repudiable evidence of receipt (EOR).

A certified email protocol is said to be fair if it guarantees that during the exchange of the items, no participant involved in the protocol can gain a significant advantage over the other participant, even if the protocol is halted for any reason. In other words, a certified email protocol needs to protect the user who is honest and guarantee the integrity and secrecy of the mail content by dishonest person. In case of dispute occurs, trusted third party should resolve effectively without causing any damage to honest participants. The communication channels between the TTP and the other agents are assumed to be resilient, i.e. all data is delivered after a finite, but unknown amount of time. The communication channels between the other agents are assumed to be unreliable, i.e. data may be lost.

In recent years, many certified email protocols that were aimed to satisfy the above properties using different types of TTP have been proposed. TTP can be in-line TTP, online TTP or off-line (optimistic) TTP. A trusted third party acting as a delivery authority, intervening in each transmission between the sender and the receiver, is called in-line TTP. Although this type of TTP can guarantee the desired properties of certified email, this can also lead to a communication and computation bottleneck. An online TTP does not handle the items to be exchanged, but is necessary in each invocation of the main protocol. M. Abadi, N. Glew, B. Horne, and B. Pinkas [1], R.Deng, L. Gong, A. Lazar, and W. Wang

[2] and J. Zhou and D. Gollmann [3] proposed systems with this type of TTP.

So trying to minimize the TTP involvement in certified email has got more attention in literature during the last years. In response to this, Off-line (optimistic) TTP that participates in the system only when disputes occur is widely used today. Although this TTP is not involved in every exchange and the sender does not need to send message via TTP, it can effectively resolve the dispute if the protocol is well designed. In this way, using offline TTP can not only provide the required cryptographic properties for certified email protocol but also reduce the delay of the message exchange process.

The proposed system is focused to develop the efficient CEM protocol with offline TTP which can guarantee the desired cryptographic strength and protect honest participants of the system.

2. Related Works

The problem of fair exchange has been studied under many different headings and from many different perspectives. For example, contract signing, key exchange and certified mail all share aspects of the problem. Several related works can be found in the literature concerning designing and verification of fair exchange protocols using different approaches. Zhiyuan Liu, Jun Pang, Chenyi Zhang [4] proposed a development of a CEM protocol with transparent TTP. They intended to be impossible to see whether TTP has been participated in the protocol or not by simply observing the evidences. In their system, TTP can not only resolve the dispute but also know the secret key which offers the confidentiality of the system.

Some common attacks against Certified Email Protocols are discovered and the countermeasures against these attacks are proposed by Min-Hua Shao, Guilin Wang, Jianying Zhou [5]. They showed the situations that the dishonest participant can get the desired item by colluding with the third participant and proposed the protocol which is resistant to this attack. Gamal A. Hussein and Fatama Helmy proposed TSRG (two stage random number generator) based certified mail service (TCMS) [6]. In their system, two-stage random number generator [7] plays a vital role to secure the transaction between the participants.

To avoid the problem that the receiver has a chance to decide whether to receive the certified email or not on the basis of the sender's identity, Nicolás González-Deleito proposed a protocol which offers the receiver the ability to receive the mail while not knowing who the sender is [8]. Enhancing Certified Email Service for Timeliness and Multicasting [9] is proposed by Jose Antonio Onieva, Jianying Zhou, Javier Lopez. Their system is aimed to reach timeliness and if the request from the receiver is out of time limit, this request is not resolved by TTP. Two generic, optimistic and efficient schemes for fair certified email deliver are proposed by Guilin Wang, Feng Bao, Kenji Imamoto and Kouichi Sakurai [10]. Their schemes provide fairness and timeliness with the help of transparent TTP. Salekul Islam and Mohammad Abu Zaid [11] analyzed and verified the key security properties of ASW protocol which is one of the prominent optimistic fair exchange protocols used for contract signing between two participants using the PRISM (Probabilistic Symbolic Model Checker) tool. The secure and confidential electronic contract signing protocol based on an on-line TTP was proposed by Antonio Ruiz-Martínez, C. Inmaculada Marín-López, Laura Baño-López and Antonio F. Gómez-Skarmeta [12]. They validate their system's security properties using AVISPA tool.

In the proposed system, a new session key is generated by the sender for each message exchange to be used in encrypting the message using the symmetric encryption algorithm and the public key encryption algorithm for encrypting the session key in order to provide the message confidentiality. This proposed email protocol is designed to be a fair and secure certified email protocol which guarantees fairness, non-repudiation, confidentiality of mail content and resistance to replay attack. Finally, the above security properties are verified by using AVISPA tool.

3. Security Enhancement of Fair Certified Email Protocol

Certified email protocols are designed to achieve fair-exchange of a message and a receipt between two potentially mistrusting parties. A protocol is said to be fair, if it is guaranteed that the receiver can get the email content if and only if the sender obtains an irrefutable receipt from the receiver. Moreover, users should be able to send important information which needs to be read only by intended receiver via email. In this paper, a CEM protocol with offline (optimistic) trusted third party that can guarantee the fair exchange of message in a fair and secret form between the sender and the receiver is proposed. To provide the secrecy of message, a new session key, generated by the sender for each message, is used as encryption key. The protocol works with three sub-protocols: exchange sub-protocol, recovery sub-protocol and abort sub-protocol.

3.1. Exchange Sub-protocol

If both participants behave honestly and send the respective items to the other properly, the message exchange procedure follows the exchange sub-protocol. The exchange sub-protocol is as follows:

1. A→B: A, B, h(k(M)), h(k), N, EOO_M

EOO_M = k_{AR}[A, B, h(k(M)), h(k), N, (k_{TU} (A, B, h(k(M)), h(k), N, k(M), k_{BU}(k))]
EOR_{M1} = k_{BR}(A, B, h(k(M)), h(k), N)
EOR_{M2} = k_{BR}(k_{TU}(A, B, h(k(M)), h(k), N, k(M), k_{BU}(k)))

EOO_M = evidence of origin of the message
EOR_M = evidence of receipt of the message
A = sender of the message
B = receiver of the message
k_{AU} = A's public key
k_{AR} = A's private key
k(M)= message encrypted with session key
h = one-way collision resistance hash function
T = Trusted Third Party (TTP)
N = nonce value

EOO_M serves as the digital signature of the message sender and EOR_M does as the digital signature of the message receiver. A digital signature is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, such that the sender cannot deny having sent the message (authentication and non-repudiation) and that the message was not altered in transit (integrity). Public key encryption algorithm (RSA) is used to generate the signature of the sender (EOO_M) and of the receiver (EOR_M) using the private keys of respective participants. Furthermore, the part intended for TTP to be used in resolution process is also consisted in EOO_M and EOR_M.

EOO_M is generated by the sender by using his own private key in order to prove that the sender sent the message. In which, not only the message that the sender wishes to send to the receiver but also the part enciphered with TTP's public key intended for TTP to be used in disputes resolution. And also EOR_M is generated by the receiver by signing upon the received evidence with his private key to be used as a proof that the receiver is actually received the message. N is intended to be used as a nonce for the purpose of protecting the replay attacks. Instead of a random value, the message sending date and time is used as the nonce value and named as N.

Hash values of the key and encrypted message (h(k(M)), h(k)) are used for the purpose of assuring integrity of the items and allowing the receiver to check whether the sender sends the true message and the key after sending EOR_M.

At first, the sender sends the IDs of the sender and receiver, the hash value of encrypted message, the nonce value (N) and the evidence of origin of the message (EOO_M) to the receiver.

2. B→A: EOR_{M1}, k_{TU} (EOR_{M2})

After checking the correctness of the evidence, the receiver generates the evidence of receipt of message (EOR_M). Then, the receiver replies the first half of EOR_M in plaintext form and the second half in encrypted form using TTP's public key so that the sender cannot get the complete evidence of receipt but first part of it before replying the ciphertext and the key at the second step of the protocol.

3. A→B: k_{AR} (k(M), k_{BU}(k))

Upon receiving the evidence from the receiver, the sender checks the signature on the first part of evidence which is not encoded. If correct, the sender sends the signed encrypted message and the key k_{AR}(k(M), k_{BU}(k)) to the receiver in turn as the third step of the protocol.

4. B→A: EOR_{M2}

After receiving this, the receiver generates a hash value from the received message and verifies the new hash value is in correspondence with the one received previously. If the message is correct, the receiver replies the plaintext form of the second half of the evidence of receipt of message to the sender. After these 4 steps, A gets the complete EOR_M to prove that B has received the message and B gets the message and EOO_M that can be used to prove that A has sent this message if dispute occurs.

3.2. Recovery Sub-protocol

The recovery procedure will be launched by one of the participants (the sender or the receiver) once the other participant misbehaves on the message or the communication channel is out of order. If the sender (A) refuses to comply in Step (3) or gives an unreasonable message or key, the receiver (B) is allowed to launch the recovery sub-protocol provided that he has sent EOR_M in step (2), but has not received the encrypted message by sending recovery request message to TTP as follows:

B → TTP : k_{BR} (EOO_M, EOR_{M1}, k_{TU} (EOR_{M2}), r)

where *r* is used to identify the recovery request.

On receipt of a message of recovery, TTP has to check the correctness of the signatures on the evidences and whether the message has been aborted or not first. If the message has been already aborted, TTP sends error message to B. If all the above checks succeed, TTP retrieves plaintext form of EOR_M, the encrypted form of message and the key from the EOO_M. TTP records the message as recovered and signs the retrieved items and sends them to the sender and the receiver respectively as follows:

TTP→B : k_{TR} (k(M), k_{BU} (k))
TTP→A : k_{TR} (EOR_{M2})

The sender is also allowed to launch the recovery protocol when the receiver fails to send EORM2 after receiving the key by sending the following recovery request message to TTP.

A→TTP : k_{AR} (EOO_M, EOR_{M1}, k_{TU} (EOR_{M2}), r)

TTP checks if the message has been aborted or not and the correctness of the signatures on the evidences. If the message has been already aborted, TTP sends the error message as before. If both checks succeed, TTP records the message as recovered and does the same resolution as in above case by sending the following items to the sender and the receiver respectively.

TTP →A : k_{TR} (EOR_{M2})
TTP →B : k_{TR} (k(M), k_{BU} (k))

After launching the recovery sub-protocol by either the sender or the receiver, both of them gets their desired items respectively.

3.3. Abort Sub-protocol

The abort sub-protocol will be launched by the sender once he is unwilling to continue the exchange protocol any more after sending the first message or the receiver failed to reply EOR_M in the second step of exchange protocol. Then A sends abort request message to TTP as follows:

A →TTP : k_{AR} (EOO_M , a)

where *a* is used to identify the abort request.

TTP verifies the signature, hash value and N on EOO_M and checks if this message is already aborted or recovered. If it is already recovered, it sends recover message to both participants and if it is already aborted, it sends abort message. If all the above checks succeed, then TTP records the message as aborted and sends abort message to both participants. After this sub-protocol, no one gets advantage over other as A did not get EOR_M from B and B did not receive the message from A too.

4. Security Analysis of the Proposed Certified Email Protocol

As it is unlikely to restrict the time that the participants should reply in email service, the dishonest participant has the time to reveal the key and the message without sending back the required items if the encrypted message is sent at the first exchange like in most protocol. In this system, the receiver cannot afford this type of cheating as the sender has to send the encrypted message along with the key only after receiving part of evidence from the receiver.

The nonce value used to be resistant to a replay attack is not needed to be generated using random number generator as the date and time that the sender starts to send the first message is used as this value. Moreover, this can provide a true random value as the sender cannot send the messages at the exact same time.

As the session key used to encrypt the mail content is sent in encoded form using the receiver's public key, no one else but only the receiver can see this key. so the secrecy of the mail content is not revealed to no one else including TTP.

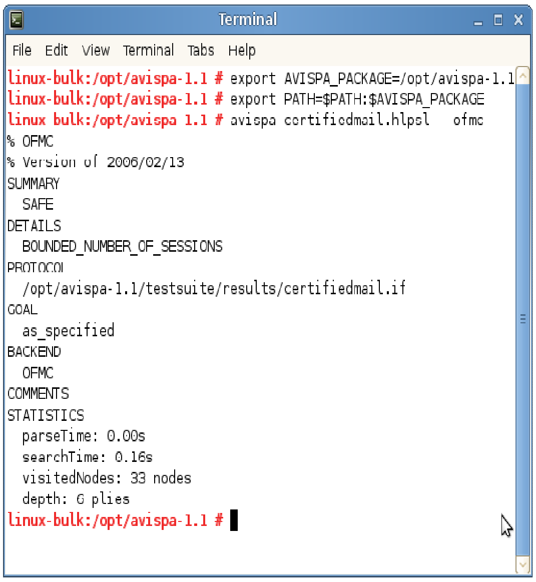
In case of a dispute, EOR_M alone from the sender is sufficient to prove that the receiver has received the message. And also the receiver is able to prove the trusted third party that the message is actually from the sender by verifying the sender’s signature on EOO_M. Since presenting (k(M), EOO_M) is sufficient for the receiver to prove that M is originally from the sender, the protocol satisfies fairness and non repudiation of origin and receipt.

In any exchange of message between the two participants, the sender can launch the abort sub-protocol after she sends out the first message. TTP will send back either an abort message or EOR_M depending on whether a recovery message has already arrived at TTP or not. The receiver can launch the resolve sub-protocol any time after receiving the first message and will get either an abort token or the ciphertext. The resilient channels between TTP, the sender and the receiver guarantees that the above procedures terminate in a timely manner.

If both the sender and receiver honestly follow the protocol and no network error occurs, only the exchange sub-protocol is run and will terminate at a state where the sender gets the desired evidence of receipt and the receiver gets both the message and the evidence of origin.

As the secret key to encrypt the message intended for the receiver is encoded with his public key, only the receiver can see the message. No one else even TTP cannot see the message content as the message contained in EOO_M ,aimed to be used by TTP to resolve the disputes, is encrypted with the secret key. So the proposed system guarantees the message confidentiality between the sender and receiver.

The AVISPA Tool is used to build and analyze the proposed certified email protocol. To describe the protocol and specify its intended properties, High Level Protocol Specification Language (HLPSL) is used which is provided by AVISPA. A translator called HLPSL2IF transforms HLPSL specifications to a low level specification with IF language (Intermediate Format) and one of four different verification backends is used to analyze the IF specifications. In this proposed system, OFMC backend is invoked to analyze if the security goals of the protocol are satisfied. The following figure shows the result of verification of the proposed protocol with AVISPA tool.



```
linux-bulk:/opt/avispa-1.1 # export AVISPA_PACKAGE=/opt/avispa-1.1
linux-bulk:/opt/avispa-1.1 # export PATH=$PATH:$AVISPA_PACKAGE
linux bulk:/opt/avispa 1.1 # avispa certifiedmail.hlpsl   ofmc
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOl
  /opt/avispa-1.1/testsuite/results/certifiedmail.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.16s
  visitedNodes: 33 nodes
  depth: 6 plies
linux-bulk:/opt/avispa-1.1 #
```

Figure 1. Verification result of the system using AVISPA tool

The output format is common to all backends of the AVISPA tool. In the SUMMARY section; it indicates if the protocol is safe, unsafe, or if the analysis is inconclusive. In a second section titled DETAILS, the tool explains under what conditions/reasons the protocol is declared safe/unsafe/inconclusive. The next sections, PROTOCOL, GOAL and BACKEND recall the name of the protocol, the goal of the analysis and the name of the backend used, respectively. Finally, some possible comments and statistics of the execution are described. Results have reported the protocol as safe, meaning that the stated security goals are successfully checked by the OFMC backend for a bounded number of sessions. Therefore, we can affirm that our protocol satisfies the security properties mentioned previously, with respect to a passive intruder.

5. Conclusion

Certified email is a value-added service to ordinary mail to send important data over the Internet with guaranteed receipt for each successful delivery. In this proposed CEM protocol, the dispute resolution process is defined because in case a dispute arises among the parties, the process must be clear enough for off-line TTP to resolve the exchange according to the evidences provided by the participating entities. And also a security analysis of the protocol using AVISPA is presented.

References

[1] M. Abadi, N. Glew, B. Horne, and B. Pinkas, "Certified email with a light on-line trusted third party: Design and implementation", *Proc. of 2002 International World Wide Web Conference (WWW'02)*, ACM press, 2002, pp. 387-395.

[2] R. Deng, L. Gong, A. Lazar, and W. Wang, "Practical protocol for certified electronic mail", *Network and Systems Management Journal*, 1996, 4(3): 279-297.

[3] J. Zhou and D. Gollmann, "Certified electronic mail", *Computer Security - ESORICS'96*, LNCS 1146, Springer-Verlag, 1996, pp. 160-171.

[4] Z. Liu, J. Pang, C. Zhang, Extending A Key-Chain Based Certified Email Protocol with Transparent TTP.

[5] M. H. Shao, G. Wang, J. Zhou, Some Common Attacks against Certified Email Protocols and the Countermeasures.

[6] G. A. Hussein, F. Helmy, TSRG based Certified Mail Service (TCMS).

[7] G. Hussein, Y. Dakroury, B. Hassen, A. Badr, Two-Stage Random Generator (TSRG); Attack-Oriented Design and Implementation, *S ´ Ecurit´edes Communications sur Internet– SECI02*, September 2002..

[8] N. González-Deleito , No Author-Based Selective Receipt in Certified Email with Tight Trust Requirements, *Proc of the 4th International Workshop for Applied PKI*.

[9] J. A. Onieva, J. Zhou, J. Lopez, Enhancing Certified Email Service for Timeliness and Multicasting.

[10] G. Wang, F. Bao, K. Imamoto, K. Sakurai, Generic, Optimistic, and Efficient Schemes for Fair Certified Email Delivery.

[11] S. Islam, M. Abu Zaid, Probabilistic Analysis and Verification of the ASW Protocol Using PRISM, *International Journal of Network Security*, Vol.7, No.3, November 2008, pp.388-396.

[12] A. Ruiz-Martínez, C. Inmaculada Marín-López, L. Baño-López, A. F. Gómez-Skarmeta, *Journal of Universal Computer Science*, Vol.15, No.3, 2009.