JOHANNES KEPLER
UNIVERSITÄT LINZ
Netzwerk für Forschung, Lehre und Praxis

Certified Mail

BACHELORARBEIT
Seminar aus Netzwerke und Sicherheit
Communication Infrastructure

zur Erlangung des akademischen Grades

Bachelor of Science (BSc)

im Bachelorstudium

INFORMATIK

Eingereicht von:
Michael Grundmann, 0656189, 521 und Andreas Wöß, 0555951, 521

Angefertigt am:
Institut für Informationsverarbeitung und Mikroprozessortechnik

Betreuung:
o.Univ.Prof. Dr. Jörg R. Mühlbacher

Mitbetreuung:
Mag. DI Dr. Andreas Putzinger

Linz, Mai, 2008

Certified Mail, the Next Step in Electronic Communication?

Michael Grundmann and Andreas Wöß

Johannes Kepler University Linz

Abstract. We describe the shortcomings of our classical electronic mail system and try to find features which an improved or certified mail system should provide. In the next part we give a possible classification of different techniques used and present some proposed protocols. We examine them according to the features and categories presented.

1 Introduction

Even though email is an increasingly important application, the Internet doesn't yet provide a reliable messaging infrastructure. Thus, an email message's sender can never be certain—and doesn't receive any evidence—that his or her message was actually delivered to and received by its intended recipients. Furthermore, a recipient can always deny having received a particular message, and the sender can't do much to prove the opposite. This lack of evidence for message delivery and reception is actually a missing piece in the infrastructure required for the more widespread and professional use of email.[1]

Our paper is organized as follows. First, we take a look at current messaging infrastructure and why it needs to be enhanced. Second, we investigate the issues which fall in the domain of certified electronic mail, starting at the most-critical ones. Third, we analyze the state of the art of non-repudiation systems. Finally, we have a look on a selected number of proposed protocols and discuss their concepts and characteristics.

1.1 Retrospection

Electronic mail, commonly referred to as *email*, is one of the most used services on the Internet, providing support to send a message to a destination. Its infrastructure was built on top of the Simple Mail Transfer Protocol (SMTP) as defined in RFC 821, issued 1982[2]; designed to be simple and effective, without security, integrity or even authentication in mind. As years passed, numerous extensions were developed to address many shortcomings of the original specification, to name only a few: Multipurpose Internet Mail Extensions (MIME)[1], authentication (RFC 2254[9]), Transport Layer Security (TLS, RFC 3207[10])

[1] MIME is specified in six linked RFC memorandum: RFC 2045, RFC 2046, RFC 2047, RFC 4288, RFC 4289 and RFC 2077[3–8], which together define the specifications.

and asymmetric cryptography extensions (for integrity and confidentiality)[2]. Still however, it provides limited mechanisms for tracking a sent message, and none for verifying that it has been delivered or read. It requires that each mail server must either deliver it onward or return a failure notice, but both software bugs and system failures can (and do) cause messages to be lost.

1.2 Why do we need certified email?

If we want a total electronic integration, we have to pass all the services offered by postal companies to the correspondent electronic version. One of these services is certified mail. Like in the paper version, certified electronic mail is one service offered to the users, such as they want to obtain a receipt (bounded to the message) from the recipient. We handle certified electronic mail as a fair exchange of values: the originator has an item (a message, and possibly a non-repudiation of origin token) to be exchanged for a recipient's item (the receipt, a non-repudiation of receipt token). The exchange has to be fair in the sense that nobody wants to send its item if they don't have the guarantee they will receive the expected item.[14]

2 Features of certified email protocols

2.1 Critical issues

A certified email protocol needs, in essence, to address at least the following issues:

Non-repudiation: The non-repudiation problem is that of preventing involved parties in a communication from falsely denying (i.e., repudiating) having taken part in that communication, and more specifically, having sent or received a particular message. This can be achieved by unforgeable digital signatures (using public key cryptography)[15]

Forms of non-repudiation: [16]

- *Non-repudiation of Origin (NRO):* Provides the recipient(s) of a message with proof of origin of the message which will protect against any attempt by the originator to falsely deny sending the message.
- *Non-repudiation of Receipt (NRR):* Provides the originator of a message with proof of receipt of the message which will protect against any attempt by the recipient(s) to falsely deny receiving the message. The provider of this service is the recipient(s).
- *Non-repudiation of Delivery (NRD):* Provides the originator of a message with proof that the message has been delivered to the originally specified recipient(s). The provider of this service is the delivery agent[3].

[2] OpenPGP: RFC 3156[11]. S/MIME: RFC 2311 and friends[12, 13].

- *Non-repudiation of Submission (NRS):* Provides the originator of a message with proof of submission of the message which will protect against any attempt by the delivery agent to falsely deny that the message was submitted for delivery to the originally specified recipient(s). Supplied by the delivery agent[3].

Fair exchange: Almost hand in hand with non-repudiation goes the concept of *fairness*: In certified mail, a message is exchanged for an acknowledgment of receipt. A simple exchange of signed items is not *fair*; it does not guard against threat of disputes, unless the signed items are exchanged simultaneously. Simultaneity, however, is hard to achieve when the parties do not interact face to face.[17, 15]

No party should be able to interrupt or corrupt the protocol to force an outcome to his/her advantage. In any instance of the protocol, it should terminate with either party having obtained the expected information, or with neither one acquiring anything. The recipient gets the mail content if and only if the mail originator receives a proof-of-receipt from the recipient. The proof-of-receipt is generally a signature that can be used to trace the transaction and certifies the mail content.[18, 19]

The classic solution to the problem of fair exchange would be to gradually exchange information over many rounds: during each round, some knowledge is revealed. If either party stops before the protocol run is complete, both parties are left with comparable knowledge and, if one assumes comparable computational capabilities, both are able to computationally recover their respective expected items of information to the same extent. The advantage of this approach is that it achieves fairness *without* a third party, but this comes at the expense of a large number of communication rounds, which is not only cumbersome, but inefficient and unrealistic in practice. Moreover, fairness depends on the assumption of approximately equal computational power which is not reasonable. Thus, to achieve an efficient protocol, the involvement of some kind of *Trusted Third Party (TTP)* is needed.[14, 18–20]

2.2 Other issues

Aside from aforementioned functionality of certified mail, a protocol should ideally additionally address the following issues:

Timeliness: Roughly speaking, timeliness guarantees that both parties will achieve their desired items in the exchange within finite time or that at least one party has the ability to decide to abort the normal operation of the protocol and adopt a scheme for protocol resolution that can be executed in a finite, eventually short, period of time. This can introduce a fairness problem, in case

[3] An inline or online Trusted Third Party (TTP), if present.

the communication channels are not reliable, i.e., always operational and without delays, because the time limit could expire before the recipient is able to reach the TTP and lead to dispute.[18] A protocol must not weaken its fairness assurance by implementing timeliness.

Efficiency: The protocol should not involve excessive computational or communication costs. It should let itself to reasonably fast implementations. Efficiency plays a particularly important role in protocols relying on the TTP to be part in every execution (which applies to inline and online TTPs), as the TTP is not only essential for the correct functioning, but also represents a potential communicational and computational bottleneck (not to mention the resulting costs). [18, 21, 22]

Confidentiality: In general, it would be desirable, especially since the delivery might involve a third party, to keep message contents delivered in private to prevent confidential information from being disclosed to intermediaries. Thus, the content should be delivered legible only to the intended recipient and sender.

Although the overhead of encrypting the message (once more) is probably almost negligible, it is not a requirement for certified email per se; it can, however, be considered certainly favorable to have it optionally available, at least. In any case it would be possible to implement another cryptographic layer on top of a certified email protocol.

Realistic trust models: The trust model should be based on realistic assumptions the users are comfortable with. A system that places less trust in outside parties is more likely to be accepted.[18]

Authenticity and Integrity: Parties involved in the protocol should be able to verify each other's identities and should not be able modify messages without such modifications being detected.[23]

Monotonicity: Each exchange of information during the protocol should add validity to the final outcome. That is, the protocol should not require any messages, certificates, or signatures to be revoked to guarantee a proper termination of the protocol. This is important, because if revocation is needed to ensure fairness, then the verification of the validity of the protocol outcome becomes a bottleneck as it requires TTP's active participation.[18]

3 State of the Art Analysis

In this section we present a classification of non-repudiation systems which we will use later to categorize some promising protocols[1, 24, 25].

3.1 Trusted Third Party

Systems without TTP On a first impression it seems that an improvement of the current protocols for electronic mail should not require any third party intervention. This could for example increase the probability of acceptance in a business environment. But such an approach has its shortcomings. As not even digital signatures are trustable without a TTP (Certificate Authority (CA) in this case), there are only a few possibilities to achieve non-repudiation. But for the purpose of certified electronic mail, none of them are usable. The main problem is fairness. A fair protocol has to ensure that a fair state is always achieved at any point. Either both parties have their desired item or none of them have. For electronic mail, this means that the sender has undeniable proof that the receiver has obtained the message. As electronic mail is an asynchronous protocol, this fair end state can only be reached through an unfair transition state in which one party has its desired item and can interrupt the process without losing it. There are protocols which try to solve this via Trusted Computing so that no party is able to get an unfair advantage. However, it is unclear when and if Trusted Computing is becoming widely deployed but the more serious issue is that such a solution would exclude everyone who does not want or is not able to switch to Trusted Computing. Furthermore, trusted systems can not yet guarantee their security or absence of error, which would be requirements in this case. Other solutions suggest a gradual release of information on both sides so that the desired item is received by both parties almost at the same time. This has two major drawbacks: First, it assumes equal processing speed of both parties. Second, it would turn electronic mail into a synchronous protocol which it was never intended to be. Therefore, non-repudiation protocols without TTP may have their fields of application but certified electronic mail is not one of them.[1, 24]

Systems with Inline TTP act in a completely different way, they do not permit any direct communication between the two parties. The TTP is included in every step of the protocol (message proxy). The TTP passes the message on to the receiver and provides the sender with a receipt. This has some implications: The TTP can read, manipulate and drop messages. Further it may give any party an unfair advantage (for example deliver the message without sending an receipt). Inspection and manipulation of the message by the TTP can be prevented by encrypting the message but in the end both parties have to completely trust that the TTP implements and executes the according protocol correctly. If we also take performance aspects into consideration the inline TTP can may easily become a performance bottleneck as it has to handle the whole load of traffic occurring. Therefore it is important to examine how much additional communication and computing effort is needed by the TTP to deliver a message and its receipt. Of course there are also advantages when using an inline TTP protocol. An inline TTP has full control over the message flow and is therefore able to provide additional services like anonymity for the sender, additional encryption, storage of the messages, etc. This centralized model also leads to more simple protocol concepts.[1, 24]

Systems with Online TTP try to address the performance problem of inline TTPs by handling all steps of the protocol except message delivery itself which takes the main load (in terms of data volume) off the TTP. Most advantages of inline TTPs still apply. The problem with all parties having to trust the TTP completely still applies, except that the TTP cannot prevent the message from being sent. To prevent an unfair situation where the receiver can deny the reception of a message the TTP may for example only hand over a decryption key for the message if the receiver admits the reception. If this results in the TTP handling the whole cryptographic process, it is again able to read and manipulate the message. Therefore some sort of multiple encryption or dual signatures has to be used to prevent this. It's an interesting aspect that protocols in this category differ quite a bit in number of necessary messages and amount of transferred data per mail. [1, 24]

Systems with Offline TTP only need to interfere in the process of exchanging message and receipt in exceptional cases (when cheating occurs or communication ceases somehow at a crucial point). In such a case the cheated party gets from the TTP what he was entitled to get from the other party. As nothing is to be gained by cheating, it can be assumed that cheating will be rare. The most obvious advantage of such a system is improved performance because the TTP is excluded as a possible bottleneck, as most exchanges can be completed directly, bypassing the TTP altogether. Such protocols are also called optimistic because they act on the assumption that the parties will behave correctly. On the other hand dispute handling is much more complicated because it may also need to take timing restraints into consideration. For example when the receiver insists that he has sent the receipt and the sender denies its reception. In case of a dispute both parties send their evidence to the TTP which then rules in favor of one party according to the specific protocol. This implies that the TTP has to be entrusted with the ability to decrypt messages without the sender's help. This again increases the level of trust needed. If the protocol also stipulates timing conditions for dispute handling (for example decryption of the message after a specific amount of time even if the sender has not admitted the reception of the receipt), it then turns into an almost synchronous protocol which is not suitable for the historically asynchronous electronic mail. [1, 24, 21]

3.2 Proposed Protocols

In this section examine some proposed protocols regarding the features described in Section 1. We also use the inclusion of a TTP for a coarse categorization.

CEM: Certified Electronic Mail as proposed in [26] CEM is designed to protect against loss and delayed delivery of mail items. Integrity protection and non-repudiation are explicitly excluded and shall be provided by an additional service.[26, 19, 25]

- Fairness: As stated above this protocol makes no attempt to protect or enforce fairness. However, the protocol may still be fair if the communication channel is always reliable and none of the parties involved acts in an unfair manner.
- Non-repudiation: in this protocol is reduced to an evidence of submission which only means that the TTP has tried to deliver the mail.
- Timeliness: As the protocol respects a deadline for the delivery report, it at least provides an in-time information on delivery failures/timeouts. But this cannot prevent that the message may have been delivered but no acknowledgment has been returned (due to a receiver acting unfair).
- Confidentiality: is respected because the TTP is given no possibility to decrypt the message. A decryption is not needed because the protocol only refers to message labels.
- Realistic trust models: As the TTP can not read the content of the messages, the parties only have to trust it to correctly handle the evidence of submission.
- TTP category: As all steps of the protocol are handled by the TTP this system clearly belongs to the inline TTP category.

A fair non-repudation Protocol [27] This protocol can be seen as an advancement of CEM[26]. It addresses fairness, non-repudiation and performance issues of the previous protocol.

- Fairness: The dependence on the fair-play of all parties and the complete reliability of the communication channel is removed. The protocol can therefore enforce fairness[27].
- Non-repudiation: Apart from the evidence of submission, the protocol also provides an evidence of receipt which fulfills our requirements stated above.
- Timeliness: The protocol provides a deadline to when messages and evidence can be retrieved.
- Confidentiality: The TTP provides an "envelope" for the message and is not able to retrieve its content.
- Realistic trust models: The parties have to trust the TTP with correct protocol handling.
- TTP category: the protocol tries to reduce the TTP's involvement, addressing possible performance issues. This suggests an Online TTP.

A Efficient non-repudiation Protocol [28] To further improve the efficiency (of [26, 27]), this protocol tries to only involve the TTP if disputes between the two parties occur. The protocol aims to be efficient if both parties play fair (which should be the normal case) and are willing to resolve conflicts themselves [28].

- Fairness: is still provided by the TTP.
- Non-repudiation: In case of dispute the TTP can act as judge and provide conflict resolution in cases of repudiation of receipt or repudiation of origin.
- Timeliness: A deadline can be chosen by the sender which refers to the TTP's clock. The recipient may reject this deadline, ending the protocol at this stage.
- Confidentiality: is not touched by this protocol variation.
- Realistic trust models: As the TTP is only involved in cases of dispute, the parties only need to trust the TTP in its role as judge.
- TTP category: The aim of the protocol clearly describes an Offline TTP system.

TRICERT: A distributed Certified E-Mail scheme[18] TRICERT adds scalability to the classic non-repudiation ideas. It introduces so called postal agents (PA) which handle the protocol on behalf of the TTP. The PAs handle the whole protocol except dispute resolution, which stays a responsibility of the TTP. Therefore only minimal trust in the PAs is required. Additional PAs may be added to distribute the load providing the promised scalability. A possible downside of this protocol is the amount of participation of the receiver that is required[18].

- Fairness: The protocol ensures fairness in all stages, even when PAs conspire with one of the parties.
- Non-repudiation: NRO is achieved because the sender has to sign the message. As no-one else is able to do that the sender cannot deny the origin of the message. NRR is also given because the recipient has to provide a receipt before he/she can retrieve the message. NRD and NRS may be provided by a PA or the TTP itself.
- Timeliness: There is a time limit in which the receiver has the possibility to complain to the TTP. The liability of the sender also ends after this time limit.
- Confidentiality: The protocol uses standard asymmetric encryption where the message is first encrypted with the receiver's public key. Therefore neither the TTP nor the PAs have access to the message content.
- Realistic trust models: Although the protocol requires full trust in the TTP, the PAs only need to be semi-trusted (similar to [29]).
- TTP category: TRICERT is a hybrid protocol which combines attributes of Online (PA) and Offline (TTP) systems.

4 Conclusion

We have gathered the common attributes for certified electronic mail mentioned in the literature on these protocols. There are quite a few proposed protocols and most of them have interesting concepts and also certain drawbacks. As we found only one implementation in use[4], we suspect that for the future a successful certified email protocol needs of course a solid scientific base but there is also much work to do when it comes to acceptability.

References

1. Oppliger, R.: Providing certified mail services on the internet. IEEE Security and Privacy (January 2007)
2. Postel, J.B.: RFC 821: Simple Mail Transfer Protocol. http://www.ietf.org/rfc/rfc821.txt (August 1982)
3. Freed, N., Borenstein, N.: RFC 2045: MIME Part One: Format of Internet Message Bodies. http://www.ietf.org/rfc/rfc2045.txt (November 1996)
4. Freed, N., Borenstein, N.: RFC 2046: MIME Part Two: Media Types. http://www.ietf.org/rfc/rfc2046.txt (November 1996)
5. Moore, K.: RFC 2047: MIME Part Three: Message Header Extensions for Non-ASCII Text. http://www.ietf.org/rfc/rfc2047.txt (November 1996)
6. Freed, N., Borenstein, N.: RFC 4288: MIME Part Four: Media Type Specifications and Registration Procedures. http://www.ietf.org/rfc/rfc4288.txt (December 2005)
7. Moore, K.: RFC 4289: MIME Part Four: Registration Procedures. http://www.ietf.org/rfc/rfc4289.txt (December 2005)
8. Freed, N., Borenstein, N.: RFC 2049: MIME Part Five: Conformance Criteria and Examples. http://www.ietf.org/rfc/rfc2049.txt (November 1996)
9. Myers, J.: RFC 2554: SMTP Service Extension for Authentication. http://www.ietf.org/rfc/rfc2554.txt (March 1999)
10. Hoffman, P.: RFC 3207: SMTP Service Extension for Secure SMTP over Transport Layer Security. http://www.ietf.org/rfc/rfc3207.txt (February 2002)
11. Freed, N., Borenstein, N.: RFC 3156: MIME Security with OpenPGP. http://www.ietf.org/rfc/rfc3156.txt (August 2001)
12. Dusse, S., Hoffman, P., Ramsdell, B., Lundblade, L., Repka, L.: RFC 2311: S/MIME Version 2 Message Specification. http://www.ietf.org/rfc/rfc2311.txt (March 1998)
13. IETF: S/MIME Working Group: S/MIME Mail Security Charter. http://www.ietf.org/html.charters/smime-charter.html (April 2008)
14. Ferrer-Gomila, J.L., Payeras-Capellà, M., i Rotger, L.H.: An efficient protocol for certified electronic mail. In: ISW '00: Proceedings of the Third International Workshop on Information Security, London, UK, Springer-Verlag (2000) 237–248
15. Louridas, P.: Some guidelines for non-repudiation protocols. SIGCOMM Comput. Commun. Rev. **30**(5) (2000) 29–38
16. Zhou, J., Gollmann, D.: Evidence and non-repudiation. J. Netw. Comput. Appl. **20**(3) (1997) 267–281
17. Asokan, N., Shoup, V., Waidner, M.: Asynchronous protocols for optimistic fair exchange. In: Proceedings of the IEEE Symposium on Research in Security and Privacy. (1998) 86–99

[4] http://www.certifiedmail.com

18. Ateniese, G., de Medeiros, B., Goodrich, M.T.: TRICERT: A distributed certified E-mail scheme, San Diego, CA, USA, Network and Distributed Systems Security Symposium (2001) 47–56
19. Markowitch, O., Gollmann, D., Kremer, S.: On fairness in exchange protocols. Information Security and Cryptology — ICISC 2002 **2587/2003** (2003) 451–465
20. Asokan, N., Schunter, M., Waidner, M.: Optimistic protocols for fair exchange. In: CCS '97: Proceedings of the 4th ACM conference on Computer and communications security, New York, NY, USA, ACM (1997) 7–17
21. Micali, S.: Simple and fast optimistic protocols for fair electronic exchange. In: PODC '03: Proceedings of the twenty-second annual symposium on Principles of distributed computing, New York, NY, USA, ACM (2003) 12–19
22. Nenadić, A., Zhang, N., Barton, S.: Fair certified e-mail delivery. In: SAC '04: Proceedings of the 2004 ACM symposium on Applied computing, New York, NY, USA, ACM (2004) 391–396
23. Khurana, H., Hahm, H.S.: Certified mailing lists. In: ASIACCS '06: Proceedings of the 2006 ACM Symposium on Information, computer and communications security, New York, NY, USA, ACM (2006) 46–58
24. Oppliger, R.: Certified mail: the next challenge for secure messaging. Commun. ACM **47**(8) (2004) 75–79
25. Cailloux, O., Gonzalez-Deleito, N., Markowitch, O.: Fairness in certified electronic mail. In: IASTED International Conference on Networks and Communication Systems, Chiang Mai, Thailand (2006) 29–31
26. Zhou, J., Gollmann, D.: Certified electronic mail. LECTURE NOTES IN COMPUTER SCIENCE (1996) 160–171
27. Zhou, J., Gollmann, D.: A fair non-repudiation protocol. IEEE Symposium on Security and Privacy **00** (1996) 0055
28. Zhou, J., Gollmann, D.: An efficient non-repudiation protocol. 10th Computer Security Foundations Workshop **00** (1997) 126
29. Franklin, M.K., Reiter, M.K.: Fair exchange with a semi-trusted third party (extended abstract). In: CCS '97: Proceedings of the 4th ACM conference on Computer and communications security, New York, NY, USA, ACM (April 1997) 1–5