Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 9, 2020
I. Symeonidis
University of Luxembourg
B. Hoeneisen
Ucom.ch
July 08, 2019

Privacy and Security Threat Analysis and Requirements for Private Messaging
draft-symeonidis-medup-requirements-00

Abstract

[RFC8280] has identified and documented important principles, such as Data Minimization, End-to-End, and Interoperability in order to enable access to fundamental Human Rights. While (partial) implementations of these concepts are already available, many current applications lack Privacy support that the average user can easily navigate. This document covers analysis of threats to privacy and security and derives requirements from this threat analysis.

Table of Contents

1. Introduction

[RFC8280] has identified and documented important principles, such as Data Minimization, End-to-End, and Interoperability in order to enable access to fundamental Human Rights. While (partial) implementations of these concepts are already available, many current applications lack Privacy support that the average user can easily navigate.

In MEDUP these issues are addressed based on Opportunistic Security [RFC7435] principles.

This documents covers analysis of threats to privacy and security and derives requirements from this threat analysis.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

1.2. Terms

The following terms are defined for the scope of this document:

o Trustwords: A scalar-to-word representation of 16-bit numbers (0 to 65535) to natural language words. When doing a Handshake, peers are shown combined Trustwords of both public keys involved to ease the comparison. [I-D.birk-pep-trustwords]

o Trust On First Use (TOFU): cf. [RFC7435], which states: "In a protocol, TOFU calls for accepting and storing a public key or credential associated with an asserted identity, without authenticating that assertion. Subsequent communication that is authenticated using the cached key or credential is secure against an MiTM attack, if such an attack did not succeed during the vulnerable initial communication."

o Man-in-the-middle (MITM) attack: cf. [RFC4949], which states: "A form of active wiretapping attack in which the attacker intercepts

and selectively modifies communicated data to masquerade as one or more of the entities involved in a communication association."

2. Motivation and Background

2.1. Objectives

- An open standard for secure messaging requirements

- Unified evaluation framework: unified goals and threat models

- Common pitfalls

- Future directions on requirements and technologies

- Misleading products on the wild (EFF secure messaging scorecard)

2.2. Known Implementations

2.2.1. Pretty Easy Privacy (pEp)

To achieve privacy of exchanged messages in an opportunistic way [RFC7435], the following model (simplified) is proposed by pEp (pretty Easy Privacy) [I-D.birk-pep]:

```
             -----                                       -----
             | A |                                       | B |
             -----                                       -----
               |                                           |
   +-------------------------+                 +-------------------------+
   | auto-generate key pair  |                 | auto-generate key pair  |
   |    (if no key yet)      |                 |    (if no key yet)      |
   +-------------------------+                 +-------------------------+
               |                                           |
   +-----------------------+                    +-----------------------+
   | Privacy Status for B: |                    | Privacy Status for A: |
   |     *Unencrypted*     |                    |     *Unencrypted*     |
   +-----------------------+                    +-----------------------+
               |                                           |
               |   A sends message to B (Public Key        |
               |   attached) / optionally signed, but      |
               |               NOT ENCRYPTED               |
               +------------------------------------------>|
               |                                           |
               |                                +-----------------------+
               |                                | Privacy Status for A: |
               |                                |      *Encrypted*      |
               |                                +-----------------------+
               |                                           |
               |      B sends message to A (Public Key     |
               |      attached) / signed and ENCRYPTED     |
               |<------------------------------------------+
               |                                           |
   +-----------------------+                               |
   | Privacy Status for B: |                               |
   |      *Encrypted*      |                               |
   +-----------------------+                               |
               |                                           |
               |   A and B successfully compare their      |
               |   Trustwords over an alternative channel  |
               |   (e.g., phone line)                      |
               |<-- -- -- -- -- -- -- -- -- -- -- -- -- -->|
               |                                           |
   +-----------------------+                    +-----------------------+
   | Privacy Status for B: |                    | Privacy Status for A: |
   |       *Trusted*       |                    |       *Trusted*       |
   +-----------------------+                    +-----------------------+
               |                                           |
```

pEp is intended to solve three problems :

- Key management
- Trust management
- Identity management

pEp is intended to be used in pre-existing messaging solutions and provide Privacy by Default, at a minimum, for message content. In addition, pEp provides technical data protection including metadata protection.

An additional set of use cases applies to enterprise environments only. In some instances, the enterprise may require access to message content. Reasons for this may include the need to conform to compliance requirements or virus/malware defense.

2.2.2. Autocrypt

Another known approach in this area is Autocrypt. Compared to pEp (cf. Section 2.2.1) - there are certain differences, for example, regarding the prioritization of support for legacy PGP [RFC4880] implementations.

More information on Autocrypt can be found on: https://autocrypt.org/background.html

[[TODO: Input from autocrypt group]]

2.3. Focus Areas (Design Challenges):

- Trust establishment: some human interaction
- Conversation security: no human interaction
- Transport privacy: no human interaction

3. System Model

3.1. Entities

- Users, sender and receiver(s): The communicating parties who exchange messages, typically referred to as senders and receivers.
- Messaging operators and network nodes: The communicating service providers and network nodes that are responsible for message delivery and synchronization.

- o Third parties: Any other entity who interacts with the messaging system.

3.2. Basic Functional Requirements

This section outlines the functional requirements. We follow the requirements extracted from the literature on private emails and instant messaging [Unger] [Ermoshina] [Clark].

- o Message: send and receive message(s)

- o Multi-device support: synchronization across multiple devices

- o Group messaging: communication of more than 2 users

[[TODO: Add more text on Group Messaging requirements.]]

4. Threat Analyses

This section describes a set of possible threats. Note that not all threats can be addressed, due to conflicting requirements.

4.1. Adversarial model

An adversary is any entity who leverages threats against the communication system, whose goal is to gain improper access to the message content and users' information. They can be anyone who is involved in communication, such as users of the system, message operators, network nodes, or even third parties.

- o Internal - external: An adversary can seize control of entities within the system, such as extracting information from a specific entity or preventing a message from being sent. An external adversary can only compromise the communication channels themselves, eavesdropping and tampering with messaging such as performing Man-in-the-Middle (MitM) attacks.

- o Local - global: A local adversary can control one entity that is part of a system, while a global adversary can seize control of several entities in a system. A global adversary can also monitor and control several parts of the network, granting them the ability to correlate network traffic, which is crucial in performing timing attacks.

- o Passive - active: A passive attacker can only eavesdrop and extract information, while an active attacker can tamper with the messages themselves, such as adding, removing, or even modifying them.

Attackers can combine these adversarial properties in a number of ways, increasing the effectiveness - and probable success - of their attacks. For instance, an external global passive attacker can monitor multiple channels of a system, while an internal local active adversary can tamper with the messages of a targeted messaging provider [Diaz].

4.2. Security Threats and Requirements

4.2.1. Spoofing and Entity Authentication

Spoofing occurs when an adversary gains improper access to the system upon successfully impersonating the profile of a valid user. The adversary may also attempt to send or receive messages on behalf of that user. The threat posed by an adversary's spoofing capabilities is typically based on the local control of one entity or a set of entities, with each compromised account typically is used to communicate with different end-users. In order to mitigate spoofing threats, it is essential to have entity authentication mechanisms in place that will verify that a user is the legitimate owner of a messaging service account. The entity authentication mechanisms typically rely on the information or physical traits that only the valid user should know/possess, such as passwords, valid public keys, or biometric data like fingerprints.

4.2.2. Information Disclosure and Confidentiality

An adversary aims to eavesdrop and disclose information about the content of a message. They can attempt to perform a man-in-the-middle attack (MitM). For example, an adversary can attempt to position themselves between two communicating parties, such as gaining access to the messaging server and remain undetectable while collecting information transmitted between the intended users. The threat posed by an adversary can be from local gaining control of one point of a communication channel such as an entity or a communication link within the network. The adversarial threat can also be broader in scope, such as seizing global control of several entities and communication links within the channel. That grants the adversary the ability to correlate and control traffic in order to execute timing attacks, even in the end-to-end communication systems [Tor]. Therefore, confidentiality of messages exchanged within a system should be guaranteed with the use of encryption schemes

4.2.3. Tampering With Data and Data Authentication

An adversary can also modify the information stored and exchanged between the communication entities in the system. For instance, an adversary may attempt to alter an email or an instant message by

changing the content of them. As a result, it can be anyone but the users who are communicating, such as the message operators, the network node, or third parties. The threat posed by an adversary can be in gaining local control of an entity which can alter messages, usually resulting in a MitM attack on an encrypted channel. Therefore, no honest party should accept a message that was modified in transit. Data authentication of messages exchanged needs to be guaranteed, such as with the use of Message Authentication Code (MAC) and digital signatures.

4.2.4. Repudiation and Accountability (Non-Repudiation)

Adversaries can repudiate, or deny, the status of the message to users of the system. For instance, an adversary may attempt to provide inaccurate information about an action performed, such as about sending or receiving an email. An adversary can be anyone who is involved in communicating, such as the users of the system, the message operators, and the network nodes. To mitigate repudiation threats, accountability, and non-repudiation of actions performed must be guaranteed. Non-repudiation of action can include proof of origin, submission, delivery, and receipt between the intended users. Non-repudiation can be achieved with the use of cryptographic schemes such as digital signatures and audit trails such as timestamps.

4.3. Privacy Threats and Requirements

4.3.1. Identifiability - Anonymity

Identifiability is defined as the extent to which a specific user can be identified from a set of users, which is the identifiability set. Identification is the process of linking information to allow the inference of a particular user's identity [RFC6973]. An adversary can identify a specific user associated with Items of Interest (IOI), which include items such as the ID of a subject, a sent message, or an action performed. For instance, an adversary may identify the sender of a message by examining the headers of a message exchanged within a system. To mitigate identifiability threats, the anonymity of users must be guaranteed. Anonymity is defined from the attackers perspective as the "the attacker cannot sufficiently identify the subject within a set of subjects, the anonymity set" [Pfitzmann]. Essentially, in order to make anonymity possible, there always needs to be a set of possible users such that for an adversary the communicating user is equally likely to be of any other user in the set [Diaz]. Thus, an adversary cannot identify who is the sender of a message. Anonymity can be achieved with the use of pseudonyms and cryptographic schemes such as anonymous remailers (i.e., mixnets), anonymous communications channels (e.g., Tor), and secret sharing.

4.3.2. Linkability - Unlinkability

Linkability occurs when an adversary can sufficiently distinguish within a given system that two or more IOIs such as subjects (i.e., users), objects (i.e., messages), or actions are related to each other [Pfitzmann]. For instance, an adversary may be able to relate pseudonyms by analyzing exchanged messages and deduce that the pseudonyms belong to one user (though the user may not necessarily be identified in this process). Therefore, unlinkability of IOIs should be guaranteed through the use of pseudonyms as well as cryptographic schemes such as anonymous credentials.

4.3.3. Detectability and Observability - Undetectability

Detectability occurs when an adversary is able to sufficiently distinguish an IOI, such as messages exchanged within the system, from random noise [Pfitzmann]. Observability occurs when that detectability occurs along with a loss of anonymity for the entities within that same system. An adversary can exploit these states in order to infer linkability and possibly identification of users within a system. Therefore, undetectability of IOIs should be guaranteed, which also ensures unobservability. Undetectability for an IOI is defined as that "the attacker cannot sufficiently distinguish whether it exists or not." [Pfitzmann]. Undetectability can be achieved through the use of cryptographic schemes such as mix-nets and obfuscation mechanisms such as the insertion of dummy traffic within a system.

4.4. Information Disclosure - Confidentiality

Information disclosure - or loss of confidentiality - about users, message content, metadata or other information is not only a security but also a privacy threat that a communicating system can face. For example, a successful MitM attack can yield metadata that can be used to determine with whom a specific user communicates with, and how frequently. To guarantee the confidentiality of messages and prevent information disclosure, security measures need to be guaranteed with the use of cryptographic schemes such as symmetric, asymmetric or homomorphic encryption and secret sharing.

4.5. Non-repudiation and Deniability

Non-repudiation can be a threat to a user's privacy for private messaging systems, in contrast to security. As discussed in section 6.1.4, non-repudiation should be guaranteed for users. However, non-repudiation carries a potential threat vector in itself when it is used against a user in certain instances. For example, whistle-blowers may find non-repudiation used against them by adversaries,

particularly in countries with strict censorship policies and in cases where human lives are at stake. Adversaries in these situations may seek to use shreds of evidence collected within a communication system to prove to others that a whistle-blowing user was the originator of a specific message. Therefore, plausible deniability is essential for these users, to ensure that an adversary can neither confirm nor contradict that a specific user sent a particular message. Deniability can be guaranteed through the use of cryptographic protocols such as off-the-record messaging.

[[TODO: Describe relation of the above introduced Problem Areas to scope of MEDUP]]

5. Specific Security and Privacy Requirements

[[This section is still in early draft state, to be substantially improved in future revisions. Among other things, there needs to be clearer distinction between MEDUP requirements, and those of a specific implementation.]]

5.1. Messages Exchange

5.1.1. Send Message

- Send encrypted and signed message to another peer

- Send unencrypted and unsigned message to another peer

  Note: Subcases of sending messages are outlined in Section 6.2.

5.1.2. Receive Message

- Receive encrypted and signed message from another peer

- Receive encrypted, but unsigned message from another peer

- Receive signed, but unencrypted message from another peer

- Receive unencrypted and unsigned message from another peer

  Note: Subcases of receiving messages are outlined in Section 6.3.

5.2. Trust Management

- Trust rating of a peer is updated (locally) when:

  * Public Key is received the first time

  * Trustwords have been compared successfully and confirmed by user (see above)

  * Trust of a peer is revoked (cf. Section 5.3, Key Reset)

- Trust of a public key is synchronized among different devices of the same user

  Note: Synchronization management (such as the establishment or revocation of trust) among a user's own devices is described in Section 5.4

5.3. Key Management

- New Key pair is automatically generated at startup if none are found.

- Public Key is sent to peer via message attachment

- Once received, Public Key is stored locally

- Key pair is declared invalid and other peers are informed (Key Reset)

- Public Key is marked invalid after receiving a key reset message

- Public Keys of peers are synchronized among a user's devices

- Private Keys are synchronized among a user's devices

  Note: Synchronization management (such as establish or revoke trust) among a user's own devices is described in Section 5.4

5.4. Synchronization Management

A device group is comprised of devices belonging to one user, which share the same key pairs in order to synchronize data among them. In a device group, devices of the same user mutually grant authentication.

- Form a device group of two (yet ungrouped) devices of the same user

- o Add another device of the same user to existing device group

- o Leave device group

- o Remove other device from device group

5.5. Identity Management

- o All involved parties share the same identity system

5.6. User Interface

[[TODO]]

6. Subcases

6.1. Interaction States

The basic model consists of different interaction states:

1. Both peers have no public key of each other, no trust possible

2. Only one peer has the public key of the other peer, but no trust

3. Only one peer has the public key of the other peer and trusts that public key

4. Both peers have the public key of each other, but no trust

5. Both peers have exchanged public keys, but only one peer trusts the other peer's public key

6. Both peers have exchanged public keys, and both peers trust the other's public key

The following table shows the different interaction states possible:

state	Peer's Public Key available	My Public Key available to Peer	Peer Trusted	Peer trusts me
1.	no	no	N/A	N/A
2a.	no	yes	N/A	no
2b.	yes	no	no	N/A
3a.	no	yes	N/A	yes
3b.	yes	no	yes	N/A
4.	yes	yes	no	no
5a.	yes	yes	no	yes
5b.	yes	yes	yes	no
6.	yes	yes	yes	yes

In the simplified model, only interaction states 1, 2, 4 and 6 are depicted. States 3 and 5 may result from e.g. key mistrust or abnormal user behavior. Interaction states 1, 2 and 4 are part of TOFU. For a better understanding, you may consult the figure in Section 2.2.1 above.

Note: In situations where one peer has multiple key pairs, or group conversations are occurring, interaction states become increasingly complex. For now, we will focus on a single bilateral interaction between two peers, each possessing a single key pair.

[[Note: Future versions of this document will address more complex cases]]

6.2. Subcases for Sending Messages

o If peer's Public Key not available (Interaction States 1, 2a, and 3a)

   * Send message Unencrypted (and unsigned)

o If peer's Public Key available (Interaction States 2b, 3b, 4, 5a, 5b, 6)

   * Send message Encrypted and Signed

6.3. Subcases for Receiving Messages

- o If peer's Public Key not available (Interaction States 1, 2a, and 3a)

  * If message is signed

    + ignore signature

  * If message is encrypted

    + decrypt with caution

  * If message unencrypted

    + No further processing regarding encryption

- o If peer's Public Key available or can be retrieved from received message (Interaction States 2b, 3b, 4, 5a, 5b, 6)

  * If message is signed

    + verify signature

    + If message is encrypted

      - Decrypt

    + If message unencrypted

      - No further processing regarding encryption

  * If message unsigned

    + If message is encrypted

      - exception

    + If message unencrypted

      - No further processing regarding encryption

7. Security Considerations

Relevant security considerations are outlined in Section 4.2.

8. Privacy Considerations

Relevant privacy considerations are outlined in Section 4.3.

9. IANA Considerations

This document requests no action from IANA.

[[RFC Editor: This section may be removed before publication.]]

10. Acknowledgments

The authors would like to thank the following people who have provided feedback or significant contributions to the development of this document: Athena Schumacher, Claudio Luck, Hernani Marques, Kelly Bristol, Krista Bennett, and Nana Karlstetter.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <https://www.rfc-editor.org/info/rfc4949>.

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <https://www.rfc-editor.org/info/rfc7435>.

11.2. Informative References

[Clark] Clark, J., van Oorschot, P., Ruoti, S., Seamons, K., and D. Zappala, "Securing Email", CoRR abs/1804.07706, 2018.

[Diaz] Diaz, C., Seys, St., Claessens, J., and B. Preneel, "Towards Measuring Anonymity", PET Privacy Enhancing Technologies, Second International Workshop, San Francisco, CA, USA, April 14-15, 2002, Revised Papers, pp. 54-68, 2002.

[Ermoshina]
Ermoshina, K., Musiani, F., and H. Halpin, "End-to-End Encrypted Messaging Protocols: An Overview", INSCI 2016: pp. 244-254, 2016.

[I-D.birk-pep]
Marques, H. and B. Hoeneisen, "pretty Easy privacy (pEp): Privacy by Default", draft-birk-pep-03 (work in progress), March 2019.

[I-D.birk-pep-trustwords]
Birk, V., Marques, H., and B. Hoeneisen, "IANA Registration of Trustword Lists: Guide, Template and IANA Considerations", draft-birk-pep-trustwords-03 (work in progress), March 2019.

[Pfitzmann]
Pfitzmann, A. and M. Hansen, "A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management", 2010, <https://nyuscholars.nyu.edu/en/publications/sok-secure-messaging>.

[RFC4880] Callas, J., Donnerhacke, L., Finney, H., Shaw, D., and R. Thayer, "OpenPGP Message Format", RFC 4880, DOI 10.17487/RFC4880, November 2007, <https://www.rfc-editor.org/info/rfc4880>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/info/rfc6973>.

[RFC8280] ten Oever, N. and C. Cath, "Research into Human Rights Protocol Considerations", RFC 8280, DOI 10.17487/RFC8280, October 2017, <https://www.rfc-editor.org/info/rfc8280>.

[Tor] Project, T., "One cell is enough to break Tor's anonymity", June 2019, <https://blog.torproject.org/one-cell-enough-break-tors-anonymity/>.

[Unger]   Unger, N., Dechand, S., Bonneau, J., Fahl, S., Perl, H., Goldberg, I., and M. Smith, "SoK: Secure Messaging", IEEE Proceedings - 2015 IEEE Symposium on Security and Privacy, SP 2015, pages 232-249, July 2015, <https://nyuscholars.nyu.edu/en/publications/sok-secure-messaging>.

Appendix A. Document Changelog

[[RFC Editor: This section is to be removed before publication]]

- o draft-symeonidis-medup-requirements-00:

    * Initial version

Appendix B. Open Issues

[[RFC Editor: This section should be empty and is to be removed before publication]]

- o Add references to used materials (in particular threat analyses part)

- o Get content from Autocrypt (Section 2.2.2)

- o Add more text on Group Messaging requirements

- o Decide on whether or not "enterprise requirement" will go to this document

Authors' Addresses

Iraklis Symeonidis
University of Luxembourg
29, avenue JF Kennedy
L-1855 Luxembourg
Luxembourg

Email: iraklis.symeonidis@uni.lu
URI: https://wwwen.uni.lu/snt/people/iraklis_symeonidis

Bernie Hoeneisen
Ucom Standards Track Solutions GmbH
CH-8046 Zuerich
Switzerland

Phone: +41 44 500 52 40
Email: bernie@ietf.hoeneisen.ch (bernhard.hoeneisen AT ucom.ch)
URI: https://ucom.ch/