Internet Engineering Task Force
Internet-Draft
Updates: 4880bis (if approved)
Intended status: Standards Track
Expires: 17 May 2021

D. Ristea, Ed.
D. Huigens
Proton Technologies AG
P. Jovanovic
University College London
13 November 2020

Symmetric-Key Cryptography in OpenPGP
draft-ietf-openpgp-symmetric-encryption

Abstract

This document defines an extension to the OpenPGP standard to support persistent symmetric keys, used for message encryption and for authentication with message authentication codes. Symmetric cryptography can be used in contexts that do not require asymmetric cryptographic algorithms, such as data storage, for improved performance, lower key sizes, and resistance to quantum computing. Symmetric algorithms already defined by the standard are re-used and this proposal introduces no additional symmetric algorithms or packet types. It extends the definition of Secret-Key Packets, Public-Key Encrypted Session Key Packets, and Signature Packets to support symmetric operations.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The OpenPGP standard RFC 4880 [RFC4880] has supported symmetric encryption for data packets using session keys since its inception, as well as symmetric encryption using password-derived keys. This proposal extends the use of symmetric cryptography by adding support for persistent symmetric keys which can be used to symmetrically encrypt session keys. This proposal uses authenticated encryption with associated data (AEAD) as proposed by RFC 4880bis [I-D.ietf-openpgp-rfc4880bis].

The OpenPGP standard supports the use of digital signatures for authentication and integrity but no similar symmetric mechanism exists in the standard. With the introduction of persistent symmetric keys, this proposals also introduces messages authentication codes (MAC) as a symmetric counterpart to digital signatures. Specifically, this proposal uses hash-based message authentication (HMAC) to ensure the integrity of stored data.

This document describes the changes required to extend the use of symmetric-key cryptography to the encryption of session keys and HMAC-based authentication.

2. Conventions Used in This Document

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 RFC 2119 [RFC2119]. Any implementation that adheres to the format and methods specified in this document is called a compliant application. Compliant applications are a subset of the broader set of OpenPGP applications described in RFC 4880 [RFC4880] and RFC 4880bis [I-D.ietf-openpgp-rfc4880bis]. Any RFC 2119 [RFC2119] keyword within this document applies to compliant applications only.

3. Motivation

When compared to asymmetric cryptography, symmetric cryptography can provide increased performance and equivalent security with shorter keys. In contexts that do not require asymmetric cryptography, such as secure data storage where the same user encrypts and decrypts data, symmetric cryptography can be used to take advantage of its benefits.

Additionally, symmetric algorithms included in OpenPGP are vulnerable to attacks possible on quantum computers [Shor]. The development of quantum-secure asymmetric cryptography is an area of active research [NIST]. Symmetric cryptography is also affected by quantum computing but to a lesser extent which can be countered using larger keys [Grover]. Quantum-secure asymmetric encryption will be required to secure communications but proactive measures to protect OpenPGP-encrypted storage can be taken by introducing persistent symmetric keys, which can be used to re-encrypt storage.

4. Persistent symmetric-key management

To allow for symmetric keys to be stored, generic symmetric algorithm values are added to list of public-key algorithms.

This document extends section 9.1. "Public-Key Algorithms" to include:

* 25: authenticated encryption with associated data (AEAD)

* 26: hash-based message authentication code (HMAC)

Allow values 25 and 26 to be used to denote the algorithm of OpenPGP keys. This enables Secret-Key Packets to hold symmetric key material.

The specific symmetric algorithm is determined by the first field in the Algorithm-Specific Fields of a symmetric key. This document extends section 5.6 to include symmetric keys:

"5.6.7 Algorithm-Specific Part for AEAD Keys

The public key information is composed of:

* A one-octet symmetric cipher

* A SHA2-256 hash of the first 32 octets of the private key fields

The private key information is composed of:

* A 32 octet random value used exclusively to generate the public hash value

* Symmetric key material of appropriate length for the chosen cipher"

"5.6.8 Algorithm-Specific Part for HMAC Keys

The public key information is composed of:

* A one-octet hash algorithm

* A SHA2-256 hash of the first 32-octet value of the private key fields

The private key information is composed of:

* A 32-octet random value used exclusively to generate the public hash value

* Symmetric key material of appropriate length for the chosen cipher"

As the secret key material is required for all cryptographic operations with symmetric keys, implementations SHOULD NOT export Public-Key Packets from Secret-Key Packets holding symmetric key material.

5. AEAD encryption of session keys

Reuse Public-Key Encrypted Session Key Packets to hold symmetrically encrypted session keys.

This document extends section "5.1. Public-Key Encrypted Session Key Packets (Tag 1)" to append the following to the list of Algorithm-Specific Fields definitions:

"Algorithm-Specific Fields for symmetric AEAD encryption:

* A one-octet AEAD algorithm

* A starting initialization vector of size specified by AEAD mode

* A symmetric key encryption of "m" performed using the selected symmetric-key cipher operating in the given AEAD mode, prefixed with a one-octet length "

For backwards compatibility, the value "m" is derived from the session key as is specified for existing algorithms, except that the PKCS #1.5 padding step is omitted.

To reflect the usage of Public-Key Encrypted Session Key Packets (Tag 1) for storing AEAD encrypted session keys, the name of Tag 1 packets is changed to Key Encrypted Session Key Packets (Tag 1).

6. HMAC-based signature packets

Save HMAC tags as digital signatures in Signature Packet (Tag 2) packets.

This document extends section "5.2.3. Version 4 and 5 Signature Packet Formats" to include:

"Algorithm-Specific Fields for HMAC signatures:

* An authentication tag of appropriate length for the hash function prefixed by a one octet length"

Although not required by HMAC, to maintain compatibility with existing signature implementations, no changes are made to section "5.2.4. Computing Signatures". HMAC tags MUST be produced from appropriately hashed data.

7. Other changes

To adapt to the addition of symmetric encryption, in Section 2.1. "Confidentiality via Encryption" read "To protect the key, it is encrypted with the receiver’s public key" as "To protect the key, it is encrypted with the receiver’s public key or a persistent symmetric key" and " 3. The session key is encrypted using each recipient’s public key" as " 3. The session key is encrypted using each recipient’s public key or a persistent symmetric key".

To adapt to the addition of MACs, 2.2. "Authentication via Digital Signature" read "The digital signature uses a hash code or message digest algorithm, and a public-key signature algorithm" as "The digital signature uses a hash code or message digest algorithm, and a public-key signature algorithm or a message authentication code algorithm" and "3. The sending software generates a signature from the hash code using the sender’s private key" as "3. The sending software generates a signature from the hash code using the sender’s private key or a persistent symmetric key".

8. Security Considerations

Security considerations are discussed throughout the document where appropriate.

9. References

9.1. Normative References

[I-D.ietf-openpgp-rfc4880bis]
Koch, W., carlson, b., Tse, R., Atkins, D., and D. Gillmor, "OpenPGP Message Format", Work in Progress, Internet-Draft, draft-ietf-openpgp-rfc4880bis-10, 31 August 2020, <https://tools.ietf.org/html/draft-ietf-openpgp-rfc4880bis-10>.

[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>.

[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>.

9.2. Informative References

[Grover]   Grover, L., "Quantum mechanics helps in searching for a needle in a haystack", DOI 10.1103/PhysRevLett.79.325, 1997, <https://arxiv.org/abs/quant-ph/9706033>.

[NIST]   NIST, "Post-Quantum Cryptography Standardization - Post-Quantum Cryptography: CSRC", 2020, <https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization>.

[Shor]   Shor, P., "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", DOI 10.1137/s0097539795293172, October 1997, <http://dx.doi.org/10.1137/S0097539795293172>.

Authors' Addresses

Dan Ristea (editor)
Proton Technologies AG
Chemin du Pre de Fleuri 3
CH-1228 Plan les Ouates
Switzerland

Daniel Huigens
Proton Technologies AG
Chemin du Pre de Fleuri 3
CH-1228 Plan les Ouates
Switzerland

Dr Philipp Jovanovic
University College London
Gower Street
London
WC1E 6BT
United Kingdom