Network
Internet-Draft
Intended status: Standards Track
Expires: 4 September 2025

P. Wouters
Aiven

3 March 2025

IKEv2 support for Child SA PFS policy notification
draft-pwouters-ipsecme-child-pfs-info-01

Abstract

This document defines the CHILD_PFS_INFO Notify Message Status Type Payload for the Internet Key Exchange Protocol Version 2 (IKEv2) to support exchanging the policy settings for the Perfect Forward Secrecy (PFS) and which Key Exchange (KE) method(s) setting of the initial Child SA that are expected to be used during Child SA rekey.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 4 September 2025.

Table of Contents

1. Introduction 2
1.1. Requirements Language 3
1.2. Payload Format 3
2. CHILD_PFS_INFO Notify Status Message Payload 3
3. Usage of the CHILD_PFS_INFO Notify 4
4. Operational Considerations 5
5. Security Considerations 5
6. Implementation Status 6
6.1. Libreswan 6
7. IANA Considerations 7
8. References 7
8.1. Normative References 7
8.2. Informative References 7
Author's Address 7

1. Introduction

The IKEv2 [RFC7296] protocol uses the Keying Exchange (KE) payload, formerly known as the Diffie-Hellman Group Transform payload to create an ephemeral IKE connection. During an IKE rekey, a new KE payload is used to create a new ephemeral IKE connection, resulting in Perfect Forward Secrecy (PFS).

A Child SA optionally uses its own PFS settings by including its own KE payload and list of acceptable Keying Exchange methods. During Child SA rekeys, KE payloads of acceptable Keying Exchange methods are exchanged to create PFS.

The Initial Exchanges establish both an IKE SA and a Child SA using the Keying Exchange method negotiated for the IKE SA. Thus, after the Initial Exchange, the peers are not aware of each others PFS requirements for the existing Child SA. It is common practise to either not perform PFS for Child SAs, or to only perform the same KE methods for both the IKE SA and all Child SAs. The situation is even more complex when Post-Quantum Key Exchange methods are used htat might contain multiple KE payloads, which might not all be desired for rekeying the Initial Child SA. It is currently not possible to know the desired PFS configuration for rekey of the initial Child SA. The peers find out about this problem only at the first Child SA rekey, which is typically 1 to 8 hours later.

This document introduces the CHILD_PFS_INFO Notify payload to exchange this information during the estaliblisment of the initial Child SA.

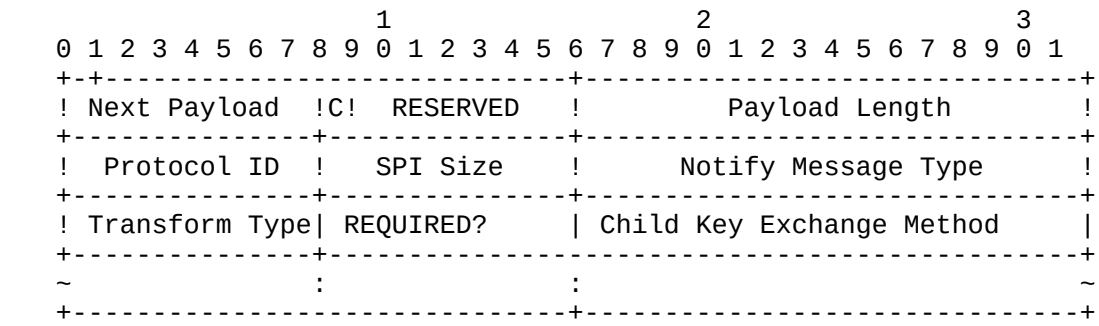
1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Payload Format

All multi-octet fields representing integers are laid out in big endian order (also known as "most significant byte first", or "network byte order").

2. CHILD_PFS_INFO Notify Status Message Payload



* The Critical Flag (C) MUST NOT be set.

* Protocol ID (1 octet) - MUST be 0. MUST be ignored if not 0.

* SPI Size (1 octet) - MUST be 0. MUST be ignored if not 0.

* Notify Status Message Type (2 octets) - set to [TBD1]

* list of one or more Child Key Exchange Methods

Each allowed or mandatory Child Key Exchange Method, and its Transform Type for which it is valid. These differ based on the Key Exchange it is used in.

Each entry is four octets. If the KE list payload is a not a multiple of four, the entire payload MUST be ignored.

The REQUIRED fields contains 0 for ALLOWED and 1 for MANDATORY.

3. Usage of the CHILD_PFS_INFO Notify

The CHILD_PFS_INFO Notify payload is sent during the (last) IKE_AUTH exchange.

Any peer MAY send the CHILD_PFS_INFO Notify payload to inform the peer of its acceptable PFS settings. If a peer receives no CHILD_PFS_INFO during the IKE_AUTH exchange, it MUST continue without any error condition. This might result in a NO_PROPOSAL_CHOSEN error during rekey time later when the initial Child SA fails to rekey.

When creating additional Child SA’s using the CREATE_CHILD_SA Exchange, the Exchange already negotiations all the required KEs and the results can be remembered to apply to future rekey events for this Child SA and CHILD_PFS_INFO MUST NOT be used.

If PFS is completely disallowed for the initial Child SA, the KE list contains only the Transform Type with value 4, the REQUIRED set to 1 (MANDATORY) and the Child Key Exchange Method set to the value 0 (NONE).

If PFS is optional for the initial Child SA but allowed, the KE list contains at least one entry for Transform Type with value 4, with one value (e.g. 19 for "256-bit random ECP group") with REQUIRED set to OPTIONAL (0).

If PFS is mandatory for the initial Child SA, the KE list contains at least one entry for Transform Type with value 4, with one value (e.g. 19 for "256-bit random ECP group") with REQUIRED set to MANDATORY (1).

To support PFS requiring additional Child Key Exchange Methods, additional allowed Child Key Exchange Methods for Additional Key Exchange Transform Types are specified that can be set to MANDATORY or OPTIONAL. Every Transform Type ID with Key Exchange Method entry in the list of Child Key Exchange Methods MUST have been used during the initial IKE SA / Child SA establishment and MUST NOT contain the value NONE (0).

Note that the Additional Key Exchange method order MUST remain the same, but the specific Transform Type number in the range 6-12 might be different if an Additional Key Exchange method was used specifically for the IKE SA but not desired for the initial Child SA rekey.

If the Child Key Exchange Method list contains any values (known or unkown) that were not used during the initial IKE SA / Child SA establishment, or any values which it is unwilling to use for PFS, it

MUST fail the Child SA. This means an Initial Responder MUST return NO_PROPOSAL_CHOSEN (and maintain the IKE SA). An Initial Initiator MUST immediately send a DELETE notify for the Child SA (not the IKE SA). This behaviour ensures that incomptabile peers will immediately fail the initial Child SA and won't only later on during rekey fail the Child SA.

4. Operational Considerations

This document is a result of Operational Considers that have shown peers can run into broken IPsec connections at rekey time. These are not obvious to the administrators as these usually do not sit around for a few hours to wait and see if the rekey process worked successfully. The CHILD_PFS_INFO results in immediate negotiation failure that can be repaired before taking the IPsec connection into production.

During rekey, the cryptographic strength of a rekeyed Child SA SHOULD remain at least as strong as the Child SA being rekeyed. In practise this means the negotiated algorithms remain the same. But some deployments use stronger settings for the IKE SA compared to its Child SAs, which means technically the initial Child SA uses a stronger KE method than for rekeys. The CHILD_PFS_INFO payload exposes such settings to the peers during the Initial Exchanges, and peers can at that time accept or reject the child proposal. Once the Initial Child SA containing CHILD_PFS_INFO is accepted, rekey proposals are guaranteed to be acceptable to both parties. For example, an IKE SA could be using KE method 15 (3072-bit MODP) and specify in the CHILD_PFS_INFO that it accepts KE method 14 (2048-bit MODP) for this Child SA rekey.

Deployments with a large number of Child SAs often use no PFS for their Child SAs. It is computationally much cheaper to establish the large number of Child SAs and then immediately rekey the IKE SA. This method can also be used if the peer's Child SA KE methods are unacceptable. If both peers accept the KE method of 0 (NONE), it can decide to rekey the Child SA without PFS and immediately rekey the IKE SA using its accepted KE method.

5. Security Considerations

This document introduces no new security considerations, as it only causes an increased awareness of peer capabilities with respect to KE methods.

6. Implementation Status

[Note to RFC Editor: Please remove this section and the reference to [RFC6982] before publication.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC7942], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

Authors are requested to add a note to the RFC Editor at the top of this section, advising the Editor to remove the entire section before publication, as well as the reference to [RFC7942].

6.1. Libreswan

Organization: The Libreswan Project

Name: https://libreswan.org/

Description: An initial IKE implementation using the Private Use value 40961 for the Notify payload

Level of maturity: Beta

Coverage: Implements the draft’s example reasons

Licensing: GPLv2

Implementation experience: TBD

Contact: Libreswan Development: swan-dev@libreswan.org

7. IANA Considerations

This document defines one new IKEv2 Notify Message Type payload for the IANA "IKEv2 Notify Message Types - Status Types" registry.

Value	Notify Type Messages - Status Types	Reference
[TBD1]	CHILD_PFS_INFO	[this document]

Figure 1

8. References

8.1. 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>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

8.2. Informative References

[RFC6982] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", RFC 6982, DOI 10.17487/RFC6982, July 2013, <https://www.rfc-editor.org/info/rfc6982>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

Author’s Address

Paul Wouters
Aiven
Email: paul.wouters@aiven.io