Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 16 November 2026

V. Smyslov
ELVIS-PLUS
T. Reddy
Nokia

15 May 2026

Separate Transports for IKE and ESP
draft-ietf-ipsecme-ikev2-reliable-transport-04

Abstract

The Internet Key Exchange protocol version 2 (IKEv2) can operate either over unreliable (UDP) transport or over reliable (TCP) transport. If TCP is used, then IPsec tunnels created by IKEv2 also use TCP. This document specifies how to decouple IKEv2 and IPsec transports so that IKEv2 can operate over TCP, while IPsec tunnels use unreliable transport. This feature allows IKEv2 to effectively exchange large blobs of data (e.g., when post-quantum algorithms are employed) while avoiding performance problems that arise when IPsec uses TCP.

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 16 November 2026.

Copyright Notice

Table of Contents

1. Introduction

The Internet Key Exchange protocol version 2 (IKEv2) [RFC7296] originally used unreliable transport (UDP) for its messages. Later it was extended to use TCP [RFC9329] where UDP is blocked. UDP remains the preferred transport for IKEv2, and TCP is only used if UDP datagrams cannot get through.

Originally IKEv2 peers exchanged only a small amount of data, so that simple retransmission mechanism on top of UDP with no congestion control sufficed. The situation has changed when post-quantum cryptographic (PQC) algorithms began to be incorporated into IKEv2 using multiple key exchanges [RFC9370]. Most post-quantum algorithms require IKE peers to exchange much more data, than classical algorithms, up to tens (or even hundreds) Kbytes. A few proposals exist that allow overcoming the 64 Kbytes limitation on the size of an IKE payload ([I-D.nir-ipsecme-big-payload], [I-D.smyslov-ipsecme-ikev2-extended-pld], [I-D.tjhai-ikev2-beyond-64k-limit]).

When IKE messages grow to tens or even hundreds of kilobytes, using UDP as a transport becomes challenging. The use of IKE fragmentation [RFC7383] helps mitigate IP fragmentation issues and ensures that each IKE message fragment fits into a UDP datagram, even if the original message does not. However, all IKE fragments are always sent (and retransmitted) simultaneously, meaning that as the number of fragments increases and congestion control remains absent, the simple retransmission mechanism of IKEv2 will perform poorly potentially causing even more problems for the network.

In some cases, a pure PQC Key Exchange may be required for specific deployments, particularly those governed by regulatory or compliance mandates that necessitate exclusive use of post-quantum cryptography. Examples include high-security environments or sectors governed by stringent cryptographic standards. In this case larger amount of data needs to be sent in the IKE_SA_INIT exchange, that makes using UDP problematic. For PQ KEM algorithms, if TCP is used for IKEv2 and peers do not require traditional algorithms, then PQ KEM can be used directly within the IKE_SA_INIT message when TCP transport is enabled for IKEv2. This approach allows IKEv2 to avoid UDP fragmentation concerns while enabling a purely post-quantum key exchange for deployments requiring exclusive PQC use.

Using reliable transport (e.g., TCP) for IKEv2 could be a solution to the problem. However, the current use of TCP for IKE and ESP [RFC9329] implies that ESP SAs are also encapsulated in TCP, which has a negative impact on IPsec performance (see Section 9 of TCP encapsulation of IKE and ESP packets [RFC9329]).

This specification allows IKE and IPsec transports to be decoupled, making it possible to use a reliable transport for IKEv2 while continuing to use an unreliable transport for IPsec.

The proposed mechanism would enable the use of all parameter sets of a post-quantum key exchange algorithm in IKE_SA_INIT as a quantum-resistant-only key exchange. This allows deployments requiring a pure post-quantum key exchange to establish keys during the IKE_SA_INIT exchange without concerns about exceeding typical network MTUs.

The idea to decouple IKE and IPsec transports was originally presented in [I-D.tjhai-ikev2-beyond-64k-limit].

2. Terminology and Notation

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.

3. Protocol Details

3.1. Discovery of TCP Transport Support for IKEv2

If the initiator supports this extension, is configured to use it, and does not know whether the responder supports IKEv2 over TCP, the initiator starts the IKE_SA_INIT exchange over UDP to responder's port 4500, as per IKEv2 [RFC7296]. In this case, the initiator includes the SEPARATE_TRANSPORTS notification (<TBA by IANA>) in the IKE_SA_INIT request. This allows the initiator to discover whether the responder supports the use of separate transports for IKE (over TCP) and ESP (over UDP). Using UDP port 4500 ensures that IPsec traffic can traverse NATs and intermediate devices that allow UDP encapsulation. If the responder has this extension enabled and receives the SEPARATE_TRANSPORTS notification in the IKE_SA_INIT request, it MUST respond with the same notification in the IKE_SA_INIT response. Upon receiving the SEPARATE_TRANSPORTS notification in the response, the initiator MUST switch to TCP port 4500 for subsequent exchanges (IKE_INTERMEDIATE or IKE_AUTH). The responder MUST be prepared to receive these exchanges over TCP. When establishing the TCP connection, the initiator MUST include the IKETCP prefix as specified in Section 3.1 of [RFC9329]. If the initiator is unable to establish a TCP connection to port 4500, it MUST cancel the current IKE SA establishment. The initiator in this case can restart the IKE_SA_INIT exchange over UDP without proposing separate transports, provided that UDP unreliability is not a critical issue for the proposed transforms.

```
Initiator (UDP)                    Responder (UDP:4500)
-----------------------------------------------------------------
IKE_SA_INIT request:
HDR , SAi1, KEi1, Ni,
[N(NAT_DETECTION_SOURCE_IP),
N(NAT_DETECTION_DESTINATION_IP),]
N(SEPARATE_TRANSPORTS)  --->
                                   IKE_SA_INIT response:
                                   HDR, SAr1, KEr1, Nr,
                                   [N(NAT_DETECTION_SOURCE_IP),
                                   N(NAT_DETECTION_DESTINATION_IP),]
                             <---  N(SEPARATE_TRANSPORTS)
```

=> Initiator switches to TCP:4500 for IKE_INTERMEDIATE /
IKE_AUTH / subsequent IKEv2 exchanges (with IKETCP prefix)
=> ESP directly over IP or ESP with UDP encapsulation - depending on the presence of NATs

3.2. Using TCP Transport for IKEv2 from the Start

Alternatively, the initiator MAY start IKE_SA_INIT over TCP port 4500 directly, as specified in TCP encapsulation of IKE and ESP packets [RFC9329], for example, when large key exchange payloads (with large public keys) are used in IKE_SA_INIT. In this case, the initiator includes the SEPARATE_TRANSPORTS notification in the IKE_SA_INIT request to indicate its preference to use separate transports; IKEv2 over TCP and ESP over UDP, provided that UDP is not blocked in the network path.

If the responder supports this extension, it includes the SEPARATE_TRANSPORTS notification in the IKE_SA_INIT response. In this case, Child SAs are created as specified in IKEv2 [RFC7296], with ESP sent over UDP (or directly over IP) if possible. If both UDP and IP are blocked, ESP is sent over TCP as described in TCP encapsulation of IKE and ESP packets [RFC9329].

If the responder does not return the SEPARATE_TRANSPORTS notification in the IKE_SA_INIT response, the initiator MUST treat this as an indication that the responder does not support separate transports. In this case, both IKEv2 messages and ESP packets MUST be sent over TCP as specified in [RFC9329].

```
Initiator (TCP)                    Responder (TCP:4500)
-------------------------------------------------------------------
IKE_SA_INIT request:
HDR , SAi1, KEi1, Ni,
[N(NAT_DETECTION_SOURCE_IP),
N(NAT_DETECTION_DESTINATION_IP),]
N(SEPARATE_TRANSPORTS)  --->
                                   IKE_SA_INIT response:
                                   HDR, SAr1, KEr1, Nr,
                                   [N(NAT_DETECTION_SOURCE_IP),
                                   N(NAT_DETECTION_DESTINATION_IP),]
                             <---  N(SEPARATE_TRANSPORTS)


=> All subsequent IKEv2 messages continue over TCP
=> ESP directly over IP or ESP with UDP encapsulation - depending on the presence of NATs
, else over TCP
```

3.3. Subsequent IKEv2 Exchanges

In both scenarios described above (Section 3.1 and Section 3.2), once the IKEv2 SA switches to TCP transport, either after IKE_SA_INIT or if TCP was used from the beginning, all subsequent IKEv2 exchanges MUST continue to use TCP. The interaction with MOBIKE is described in Section 4.1.

3.4. Notification Format

The SEPARATE_TRANSPORTS notification has Protocol ID set to 0 and SPI Size set to 0. This specification does not define any notification data, the notification is sent with no data. Future specifications may define data for this notification. Peers conforming to this specification MUST ignore any data if present.

3.5. ESP Behavior

Child SAs are created as specified in IKEv2 [RFC7296]. ESP packets either use direct transport over IP or are UDP encapsulated if NAT is detected. If UDP transport for ESP becomes unavailable (e.g., blocked by a firewall), peers MAY switch ESP to use TCP transport as specified in [RFC9329]. If ESP is transported over a different protocol than IKE, intermediate devices might apply different filtering rules. To detect possible connectivity issues with ESP traffic, the encrypted ESP ping mechanism defined in [I-D.ietf-ipsecme-encrypted-esp-ping] MAY be used.

3.6. NAT Considerations

When separate transports are used for IKEv2 and ESP, NAT traversal for each transport must be handled independently, as intermediate devices maintain NAT state per transport.

NAT detection follows the standard mechanism defined in Section 2.23 of [RFC7296]. The initiator SHOULD include NAT_DETECTION_SOURCE_IP and NAT_DETECTION_DESTINATION_IP notifications in IKE_SA_INIT, regardless of whether IKE_SA_INIT is sent over UDP or TCP. NAT detection MAY be omitted only if it is known by other means that no NAT is present on the path between the peers. If a NAT is detected, ESP MUST use UDP encapsulation on port 4500 [RFC3948]. Peers MUST maintain NAT mappings for the ESP path by sending NAT keepalive packets as specified in Section 2.23 of [RFC7296], and MUST NOT assume that the TCP connection used for IKEv2 provides any keepalive benefit for the ESP UDP path.

3.7. UDP Reachability Verification

When IKEv2 starts over UDP (Section 3.1), the successful exchange of IKE_SA_INIT messages implicitly demonstrates that UDP is reachable and NAT detection results can be used to determine whether ESP should be sent directly over IP or UDP encapsulated. No additional verification is needed.

When IKEv2 starts over TCP (Section 3.2), there is no implicit evidence that ESP traffic is reachable. After the Child SA is established, the initiator SHOULD verify ESP reachability, unless it has other means to do so (for example, the presence of incoming ESP traffic). If ESP reachability cannot be confirmed, the initiator MUST delete the current IKE SA (with DELETE payload) and re-establish it over TCP without proposing separate transport for ESP, as specified in [RFC9329]. The ESP transport to probe is determined by the NAT detection results as follows:

* If a NAT was detected, the initiator SHOULD probe ESP using UDP encapsulation on port 4500 [RFC3948].

* If no NAT was detected, the initiator SHOULD probe ESP directly over IP. If no response is received after a short delay, the initiator SHOULD also probe using UDP encapsulation on port 4500 [RFC3948], since some middleboxes do not allow IP traffic without UDP or TCP transport. The initiator MUST use the transport for which a response is received first. This approach is analogous to the Happy Eyeballs algorithm [RFC8305], giving preference to ESP sent directly over IP while avoiding excessive delay if it is not reachable.

One way to perform this verification is to use an ESP Echo Request [I-D.ietf-ipsecme-encrypted-esp-ping].

4. Interaction with IKEv2 Extensions

4.1. Interaction with MOBIKE

MOBIKE [RFC4555] allows an IKE SA, along with its Child SAs, to migrate from one IP address to another. Section 7.1 of TCP encapsulation of IKE and ESP packets [RFC9329] specifies that when using TCP as the IKE transport, a peer should attempt to switch back to UDP in the event of an IP address change. This specification updates that requirement: when separate transports are used for IKE and ESP, peers MUST NOT attempt to switch the IKE SA transport from TCP to UDP. However, an ESP SA MAY switch from UDP to TCP if UDP is blocked at the new IP address.

Similarly, when ESP is running over TCP and the initiator detects an IP address change, the initiator MUST perform UDP reachability verification as described in Section 3.7 on the new path. If ESP reachability is confirmed, the ESP SA switches from TCP to the verified path.

4.2. Interaction with IKE Session Resumption

IKE session resumption [RFC5723] allows peers to quickly re-establish an IKE SA after the connection is broken. Since network conditions may change while the client is inactive, the use of separate transports MUST NOT be stored in the resumption ticket and MUST be re-negotiated during session resumption. When resuming an IKE session, the initiator MUST start with UDP to destination port 4500, unless it is configured to use only TCP. This is because the IKE_SESSION_RESUME exchange does not transfer large public keys.

5. Security Considerations

Section 10 of TCP encapsulation of IKE and ESP packets [RFC9329] discusses security implications of using TCP as IKE transport.

6. IANA Considerations

This document defines a new Notify Message Type in the "IKEv2 Notify Message Status Types" registry:

<TBA>	SEPARATE_TRANSPORTS

7. IAcknowledgments

Thanks to Hannes Tschofenig, Dan Wing and Andrew Cagney for the discussion and comments.

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>.

[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>.

[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>.

[RFC9329] Pauly, T. and V. Smyslov, "TCP Encapsulation of Internet Key Exchange Protocol (IKE) and IPsec Packets", RFC 9329, DOI 10.17487/RFC9329, November 2022, <https://www.rfc-editor.org/info/rfc9329>.

[RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, DOI 10.17487/RFC3948, January 2005, <https://www.rfc-editor.org/info/rfc3948>.

[RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", RFC 4555, DOI 10.17487/RFC4555, June 2006, <https://www.rfc-editor.org/info/rfc4555>.

8.2. Informative References

[RFC5723] Sheffer, Y. and H. Tschofenig, "Internet Key Exchange Protocol Version 2 (IKEv2) Session Resumption", RFC 5723, DOI 10.17487/RFC5723, January 2010, <https://www.rfc-editor.org/info/rfc5723>.

[RFC8305] Schinazi, D. and T. Pauly, "Happy Eyeballs Version 2: Better Connectivity Using Concurrency", RFC 8305, DOI 10.17487/RFC8305, December 2017, <https://www.rfc-editor.org/info/rfc8305>.

[RFC7383] Smyslov, V., "Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation", RFC 7383, DOI 10.17487/RFC7383, November 2014, <https://www.rfc-editor.org/info/rfc7383>.

[RFC9370] Tjhai, CJ., Tomlinson, M., Bartlett, G., Fluhrer, S., Van Geest, D., Garcia-Morchon, O., and V. Smyslov, "Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 9370, DOI 10.17487/RFC9370, May 2023, <https://www.rfc-editor.org/info/rfc9370>.

[I-D.nir-ipsecme-big-payload] Nir, Y., "A Larger Internet Key Exchange version 2 (IKEv2) Payload", Work in Progress, Internet-Draft, draft-nir-ipsecme-big-payload-07, 17 March 2026, <https://datatracker.ietf.org/doc/html/draft-nir-ipsecme-big-payload-07>.

[I-D.tjhai-ikev2-beyond-64k-limit] Tjhai, C., Heider, T., and V. Smyslov, "Beyond 64KB Limit of IKEv2 Payloads", Work in Progress, Internet-Draft, draft-tjhai-ikev2-beyond-64k-limit-03, 28 July 2022, <https://datatracker.ietf.org/doc/html/draft-tjhai-ikev2-beyond-64k-limit-03>.

[I-D.smyslov-ipsecme-ikev2-extended-pld] Smyslov, V., "Extended IKEv2 Payload Format", Work in Progress, Internet-Draft, draft-smyslov-ipsecme-ikev2-extended-pld-01, 6 March 2023, <https://datatracker.ietf.org/doc/html/draft-smyslov-ipsecme-ikev2-extended-pld-01>.

[I-D.ietf-ipsecme-encrypted-esp-ping] Antony, A. and S. Klassert, "Encrypted ESP Echo Protocol", Work in Progress, Internet-Draft, draft-ietf-ipsecme-encrypted-esp-ping-03, 4 May 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-ipsecme-encrypted-esp-ping-03>.

Authors' Addresses

Valery Smyslov
ELVIS-PLUS
Russian Federation
Email: svan@elvis.ru

Tirumaleswar Reddy
Nokia
India
Email: kondtir@gmail.com