IP Security Maintenance and Extensions (IPsecME) WG

IETF 111, Monday, July 26th, 2021

Chairs: Tero Kivinen
Yoav Nir

Responsible AD: Benjamin Kaduk

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

• By participating in the IETF, you agree to follow IETF processes and policies.

• If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.

• As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.

• Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.

• As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:
•BCP 9 (Internet Standards Process)
•BCP 25 (Working Group processes)
•BCP 25 (Anti-Harassment Procedures)
•BCP 54 (Code of Conduct)
•BCP 78 (Copyright)
•BCP 79 (Patents, Participation)
•https://www.ietf.org/privacy-policy/ (Privacy Policy)

Administrative Tasks

Bluesheets

We need volunteers to be:

- Two note takers
- One jabber scribe

Jabber: xmpp:ipsecme@jabber.ietf.org?join

MeetEcho: https://meetings.conf.meetecho.com/ietf111/?group=ipsecme&short=&item=1

Notes: https://codimd.ietf.org/notes-ietf-111-ipsecme

Agenda

- Note Well, technical difficulties and agenda bashing – Chairs (5 min) (21:30-21:35)
- Document Status – Chairs (5 min) (21:35-21:40)
- Work items
  - Hybrid IKEv2 Interoperability Testing – Valery Smyslov (5 min) (21:40-21:45)
  - Improvements for Post-Quantum IKEv2 – Daniel Herzinger (5 min) (21:45-21:50)
  - Management of IPTFS (Yang and SNMP drafts) – Christian Hopps (10 min) (21:50-22:00)
- New items
  - IKEv2 Configuration for Encrypted DNS – Valery Smyslov (10 min) (22:00-22:10)
  - Beyond 64kB limit of IKEv2 Payloads – Valery Smyslov (10 min) (22:10-22:20)
  - IKEv2 Optional SA & TS Payloads in Child Exchange – William Panwei (5 min) (22:20-22:25)
  - IKEv2 Support for Per-Queue Child SAs – Paul Wouters (5 min) (22:25:22:30)
- AOB + Open Mic (0 min) (22:30-22:30)

WG Status Report

Waiting for write-up / Chair review:

- draft-ietf-ipsecme-ikev2-intermediate
- draft-hopps-ipsecme-iptfs

In WGLC:

- draft-ietf-ipsecme-ikev1-algo-to-historic

Work in progress:

- draft-ietf-ipsecme-g-ikev2
- draft-ietf-ipsecme-ikev2-multiple-ke
- draft-ietf-ipsecme-labeled-ipsec
- draft-smyslov-ipsecme-rfc8229bis
- draft-fedyk-ipsecme-yang-iptfs
- draft-ietf-ipsecme-mib-iptfs

Presentations

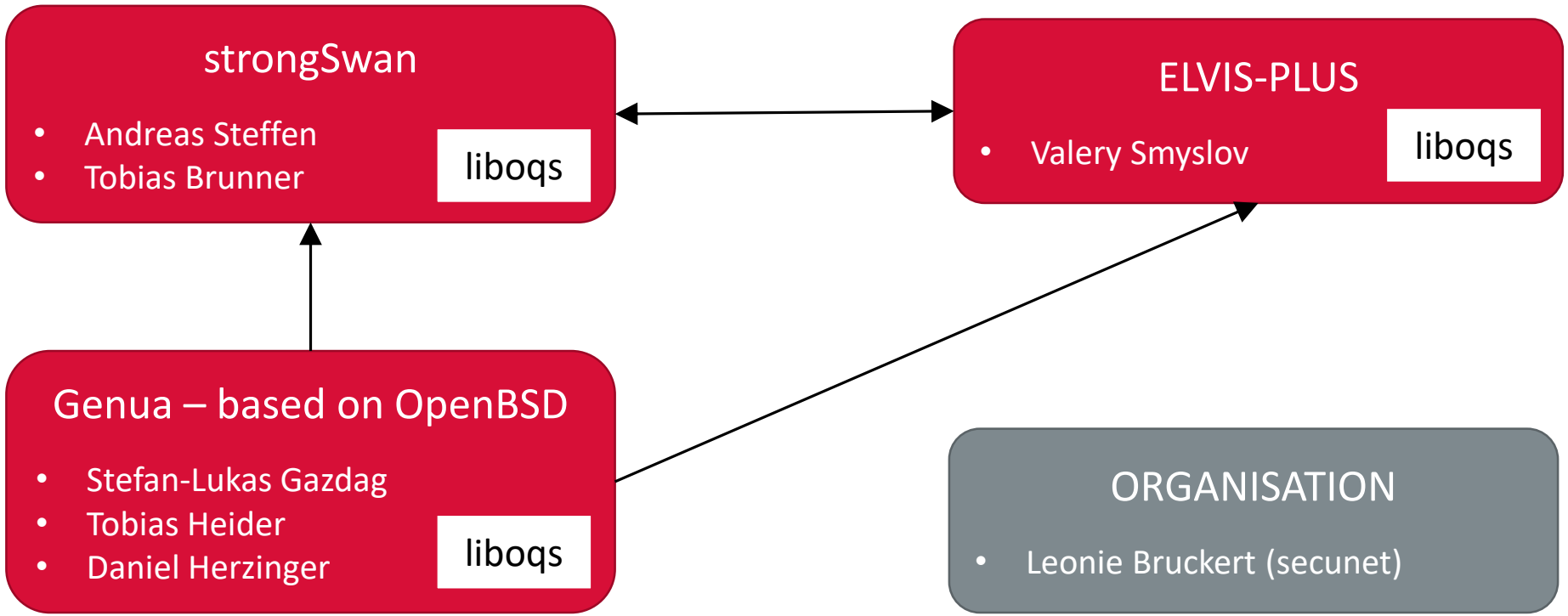
- **Hybrid IKEv2 Interoperability Testing – Valery Smyslov**
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- IKEv2 support for per-queue Child SAs – Paul Wouters

Hybrid IKEv2 Interoperability Testing

July 1st, 2021

Setting

"Hybrid IKEv2" = draft-ietf-ipsecme-ikev2-intermediate + draft-ietf-ipsecme-ikev2-multiple-ke



Results

strongSwan ⟷ ELVIS-PLUS

Successfully tested:

- Negotiation of algorithms
- Single intermediate exchange
- Up to four intermediate exchanges
- IKE SA rekeying
- Child SA rekeying
- PQC KEMs: Kyber, Saber, FrodoKEM, SIKE

Issues:

- Same algorithm selected for all three additional key exchange rounds

Genua → ELVIS-PLUS

Successfully tested:

- Negotiation of algorithms

Issues:

- (Genua) Crashing probably related to liboqs usage

Genua → strongSwan

Successfully tested:

- Negotiation of algorithms

Issues:

- (Genua) Crashing probably related to liboqs usage

Results (cont.)

- It would be useful to agree on algorithm IDs in private range until final NIST Standards are available
  → strongSwan IDs
- Further discussions in small group of interested people
- Genua and ELVIS-PLUS interested in testing beyond 64KB limit (Classic McEliece)

```
/** NIST round 3 KEM candidates, in PRIVATE USE */
KE_KYBER_L1         = 1050,
KE_KYBER_L3         = 1051,
KE_KYBER_L5         = 1052,
KE_NTRU_HPS_L1      = 1053,
KE_NTRU_HPS_L3      = 1054,
KE_NTRU_HPS_L5      = 1055,
KE_NTRU_HRSS_L3     = 1056,
KE_SABER_L1         = 1057,
KE_SABER_L3         = 1058,
KE_SABER_L5         = 1059,
/** NIST alternative KEM candidates, in PRIVATE USE */
KE_BIKE_L1          = 1060,
KE_BIKE_L3          = 1061,
KE_BIKE_L5          = 1062,
KE_FRODO_AES_L1     = 1063,
KE_FRODO_AES_L3     = 1064,
KE_FRODO_AES_L5     = 1065,
KE_FRODO_SHAKE_L1   = 1066,
KE_FRODO_SHAKE_L3   = 1067,
KE_FRODO_SHAKE_L5   = 1068,
KE_HQC_L1           = 1069,
KE_HQC_L3           = 1070,
KE_HQC_L5           = 1071,
KE_SIKE_L1          = 1072,
KE_SIKE_L2          = 1073,
KE_SIKE_L3          = 1074,
KE_SIKE_L5          = 1075,
```

Presentations

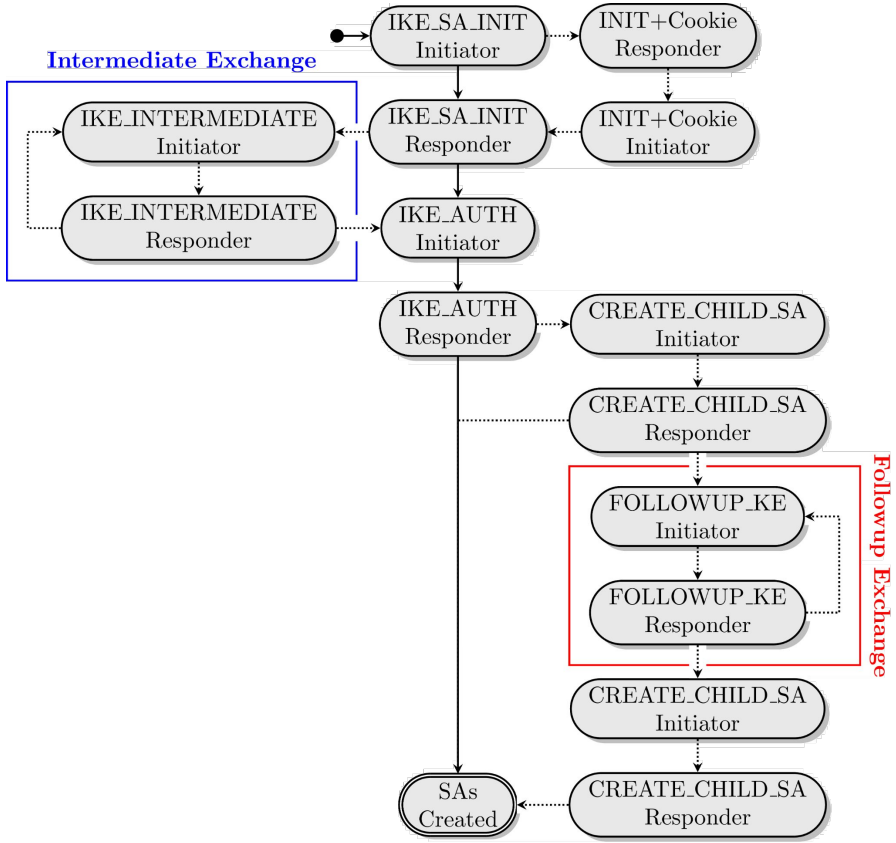
- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- **Improvements for post-quantum IKEv2 – Daniel Herzinger**
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- IKEv2 support for per-queue Child SAs – Paul Wouters

Improvements for Post-Quantum IKEv2

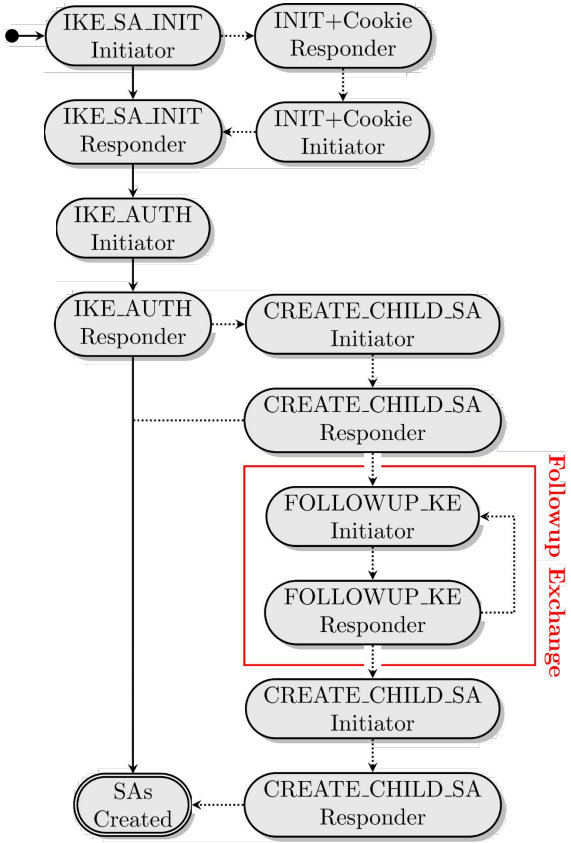
Intermediate Exchanges

- Often fragmented, especially when used for PQKEs
- Signing fragmented intermediate messages is complex
- Large PQKEs in intermediate messages lead to vulnerabilities againts DOS attacks ⇒ Move them to FOLLOWUP_KE

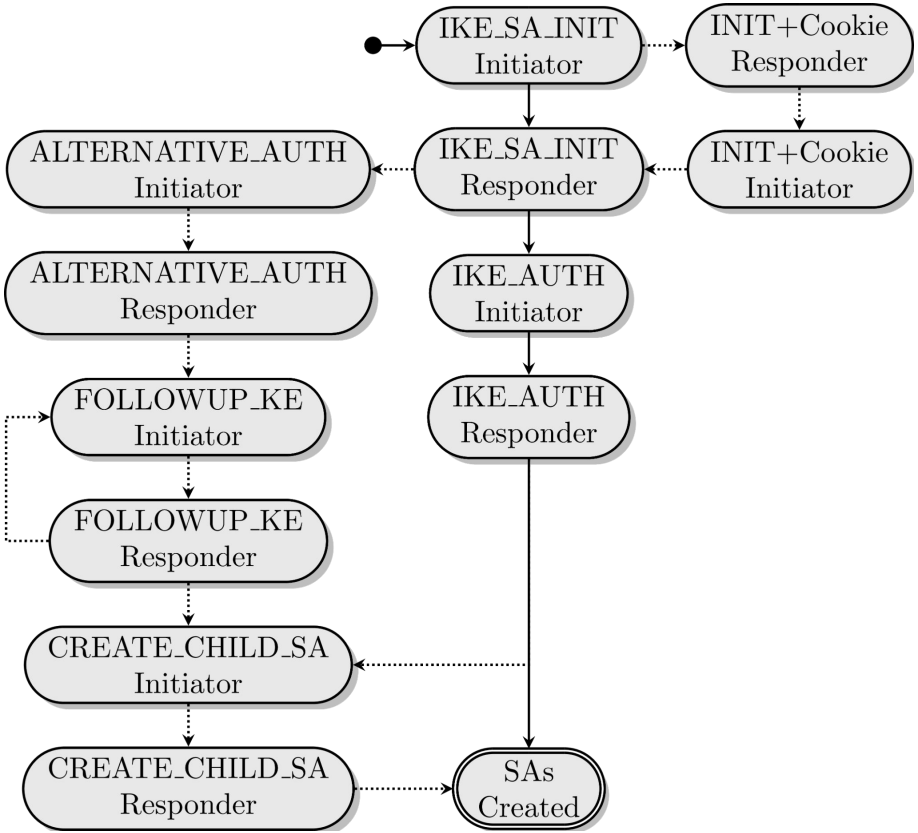
Status Quo


IKE_SA_INIT
Initiator
INIT+Cookie
Responder
Intermediate Exchange
IKE_INTERMEDIATE
Initiator
IKE_SA_INIT
Responder
INIT+Cookie
Initiator
IKE_INTERMEDIATE
Responder
IKE_AUTH
Initiator
IKE_AUTH
Responder
CREATE_CHILD_SA
Initiator
CREATE_CHILD_SA
Responder
FOLLOWUP_KE
Initiator
FOLLOWUP_KE
Responder
Followup Exchange
CREATE_CHILD_SA
Initiator
SAs
Created
CREATE_CHILD_SA
Responder

First Step: No IKE_INTERMEDIATE



Second Step: New Authentication Exchange



Benefits

- Independent of complex IKE_INTERMEDIATE exchanges
- DoS protected introduction of large PQKEs
- Clean state machine
- New place for exchanging handshake data after establishing an authenticated channel

Presentations

- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- **Management of IPTFS (Yang and SNMP draft) – Christian Hopps**
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- IKEv2 support for per-queue Child SAs – Paul Wouters

Donald Fedyk
Christian Hopps
LabN Consulting, LLC

YANG Model for IP Traffic Flow Security

IETF 111 – “draft-ietf-ipsecme-yang-iptfs-00”

No Changes since IETF110

- Base spec draft-ipsecme-iptfs-09.txt progressing since WG last call
- Augmented YANG *draft-ietf-i2nsf-sdn-ipsec-flow-protection now RFC 9061*

Next Steps

- Asking for WG last call

Current Tree (ike version shown)

```
module: ietf-ipsecme-iptfs
    augment /nsfike:ipsec-ike/nsfike:conn-entry/nsfike:spd
              /nsfike:spd-entry/nsfike:ipsec-policy-config
              /nsfike:processing-info/nsfike:ipsec-sa-cfg:
      +--rw traffic-flow-security
         +--rw congestion-control?     boolean
         +--rw packet-size
         |  +--rw use-path-mtu-discovery?   boolean
         |  +--rw outer-packet-size?        uint16
         +--rw (tunnel-rate)?
         |  +--:(l2-fixed-rate)
         |  |  +--rw l2-fixed-rate?    uint64
         |  +--:(l3-fixed-rate)
         |     +--rw l3-fixed-rate?    uint64
         +--rw dont-fragment?          boolean
         +--rw max-aggregation-time?   decimal64
    augment /nsfike:ipsec-ike/nsfike:conn-entry/nsfike:child-sa-info:
      +--ro traffic-flow-security
         +--ro congestion-control?     boolean
         +--ro packet-size
         |  +--ro use-path-mtu-discovery?   boolean
         |  +--ro outer-packet-size?        uint16
         +--ro (tunnel-rate)?
         |  +--:(l2-fixed-rate)
         |  |  +--ro l2-fixed-rate?    uint64
         |  +--:(l3-fixed-rate)
         |     +--ro l3-fixed-rate?    uint64
         +--ro dont-fragment?          boolean
```

```
 augment /nsfike:ipsec-ike/nsfike:conn-
entry/nsfike:child-sa-info:
       +--ro ipsec-stats {ipsec-stats}?
       |  +--ro tx-pkts?        uint64
       |  +--ro tx-octets?      uint64
       |  +--ro tx-drop-pkts?   uint64
       |  +--ro rx-pkts?        uint64
       |  +--ro rx-octets?      uint64
       |  +--ro rx-drop-pkts?   uint64
       +--ro iptfs-inner-pkt-stats {iptfs-stats}?
       |  +--ro tx-pkts?              uint64
       |  +--ro tx-octets?            uint64
       |  +--ro rx-pkts?              uint64
       |  +--ro rx-octets?            uint64
       |  +--ro rx-incomplete-pkts?   uint64
       +--ro iptfs-outer-pkt-stats {iptfs-stats}?
          +--ro tx-all-pad-pkts?       uint64
          +--ro tx-all-pad-octets?     uint64
          +--ro tx-extra-pad-pkts?     uint64
          +--ro tx-extra-pad-octets?   uint64
          +--ro rx-all-pad-pkts?       uint64
          +--ro rx-all-pad-octets?     uint64
          +--ro rx-extra-pad-pkts?     uint64
          +--ro rx-extra-pad-octets?   uint64
          +--ro rx-errored-pkts?       uint64
          +--ro rx-missed-pkts?        uint64
```

Donald Fedyk
Eric Kinzie
LabN Consulting, LLC

Definitions of Managed Objects for IP Traffic Flow Security

IETF 111 – “draft-ietf-ipsecme-mib-iptfs-00”

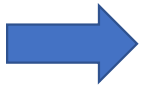
Objective: Provide a read only SNMP MIB

- Some operators still require read-only SNMP support
- Mechanically derived from the YANG model
- Adopted by WG

Next Steps

- Asking for WG last call

```
leaf l2-fixed-rate {
        type uint64;
        description
          "Target bandwidth/bit rate in bps for iptfs tunnel. This
           fixed rate is the nominal timing for the fixed size packet.
           If congestion control is enabled the rate may be adjusted
           down (or up if unset).";
        reference
          "draft-ietf-ipsecme-iptfs section 4.1";
      }
```



```
l2FixedRate OBJECT-TYPE
        SYNTAX      Counter64
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "TFS bit rate may be specified at layer 2 wire rate.
          Target bandwidth/bit rate in bps for iptfs tunnel.
          This rate is the nominal timing for the fixed size
          packet. If congestion control is enabled the rate may
          be adjusted down (or up if unset)."
        ::= { iptfsConfigTableEntry 5 }
```

MIB Tree

```
---- iptfsMIB(1.3.6.1.3.500)
    +---- iptfsMIBObjects(1)
    |  +---- iptfsGroup(1)
    |  |  +---- iptfsConfigTable(1)
    |  |     +---- iptfsConfigTableEntry(1) [iptfsConfigSaIndex]
    |  |        +---- iptfsConfigSaIndex(1) Integer32
    |  |        +--r- congestionControl(2)  TruthValue
    |  |        +--r- usePathMtu(3)         TruthValue
    |  |        +--r- outerPacketSize(4)    UnsignedShort
    |  |        +--r- l2FixedRate(5)        Counter64
    |  |        +--r- l3FixedRate(6)        Counter64
    |  |        +--r- dontFragment(7)       TruthValue
    |  |        +--r- maxAggregationTime(8) NanoSeconds
    |  +---- ipsecStatsGroup(2)
    |  |  +---- ipsecStatsTable(1)
    |  |     +---- ipsecStatsTableEntry(1) [ipsecSaIndex]
    |  |        +---- ipsecSaIndex(1)  Integer32
    |  |        +--r- txPackets(2)     Counter64
    |  |        +--r- txOctets(3)      Counter64
    |  |        +--r- txDropPackets(4) Counter64
    |  |        +--r- rxPackets(5)     Counter64
    |  |        +--r- rxOctets(6)      Counter64
    |  |        +--r- rxDropPackets(7) Counter64
    |  +---- iptfsInnerStatsGroup(3)
    |  |  +---- iptfsInnerStatsTable(1)
    |  |     +---- iptfsInnerStatsTableEntry(1) [iptfsInnerSaIndex]
    |  |        +---- iptfsInnerSaIndex(1)        Integer32
    |  |        +--r- txInnerPackets(2)           Counter64
    |  |        +--r- txInnerOctets(3)            Counter64
    |  |        +--r- rxInnerPackets(4)           Counter64
    |  |        +--r- rxInnerOctets(5)            Counter64
    |  |        +--r- rxIncompleteInnerPackets(6) Counter64
    |  +---- iptfsOuterStatsGroup(4)
    |     +---- iptfsOuterStatsTable(1)
    |        +---- iptfsOuterStatsTableEntry(1) [iptfsSaIndex]
    |           +---- iptfsSaIndex(1)      Integer32
    |           +--r- txExtraPadPackets(2) Counter64
    |           +--r- txExtraPadOctets(3)  Counter64
    |           +--r- txAllPadPackets(4)   Counter64
    |           +--r- txAllPadOctets(5)    Counter64
    |           +--r- rxExtraPadPackets(6) Counter64
    |           +--r- rxExtraPadOctets(7)  Counter64
    |           +--r- rxAllPadPackets(8)   Counter64
    |           +--r- rxAllPadOctets(9)    Counter64
    |           +--r- rxErroredPackets(10) Counter64
    |           +--r- rxMissedPackets(11)  Counter64
    +---- iptfsMIBConformance(2)
       +---- iptfsMIBConformances(1)
       |  +---- iptfsMIBCompliance(1)
       +---- iptfsMIBGroups(2)
          +---- iptfsMIBConfGroup(1)
          +---- ipsecStatsConfGroup(2)
          +---- iptfsInnerStatsConfGroup(3)
          +---- iptfsOuterStatsConfGroup(4)
```

Comments / Questions?

Presentations

- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- **Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov**
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- IKEv2 support for per-queue Child SAs – Paul Wouters

IKEv2 Configuration for Encrypted DNS

draft-btw-add-ipsecme-ike-03

IETF#111, July 2021

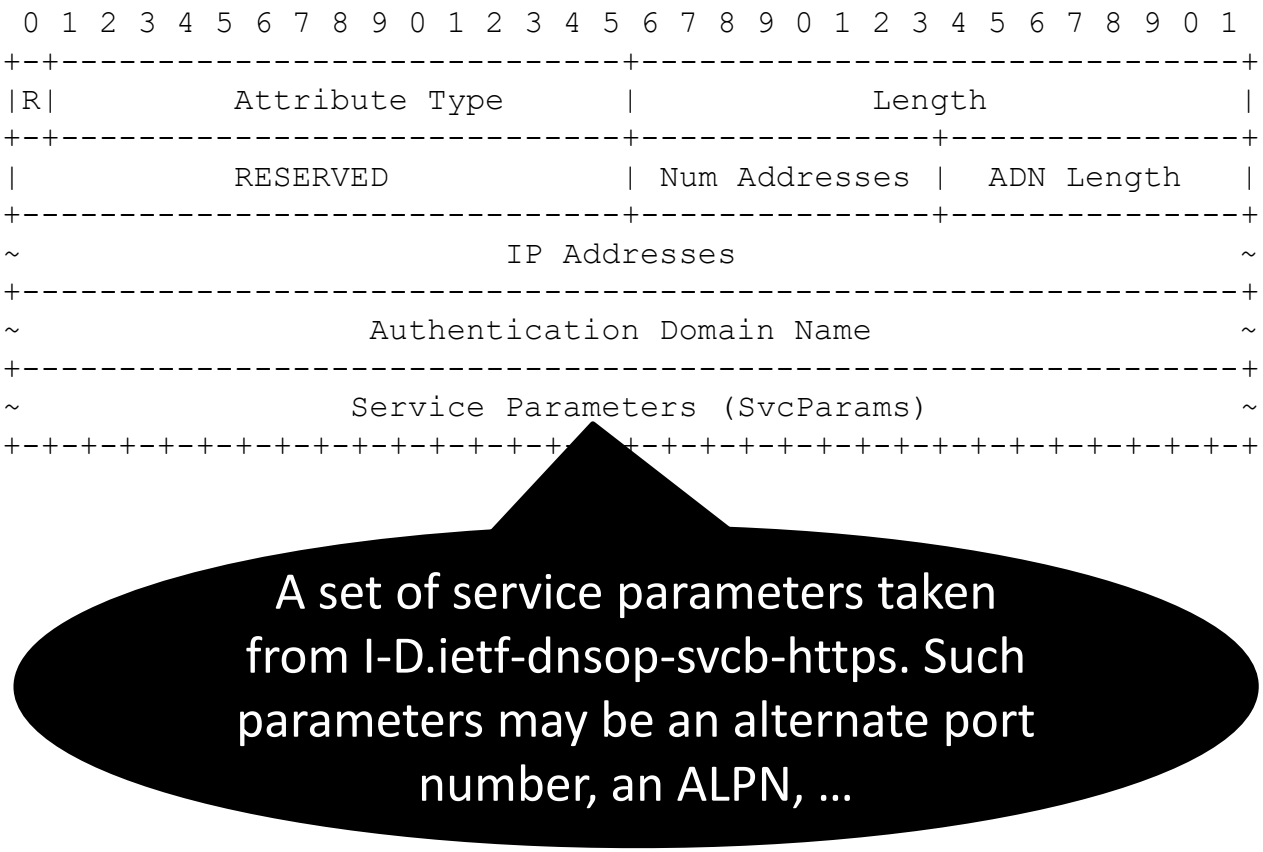
M. Boucadair (Orange)
T. Reddy (Akamai)
D. Wing (Citrix)
V. Smyslov (ELVIS-PLUS)

Changes Since IETF#110

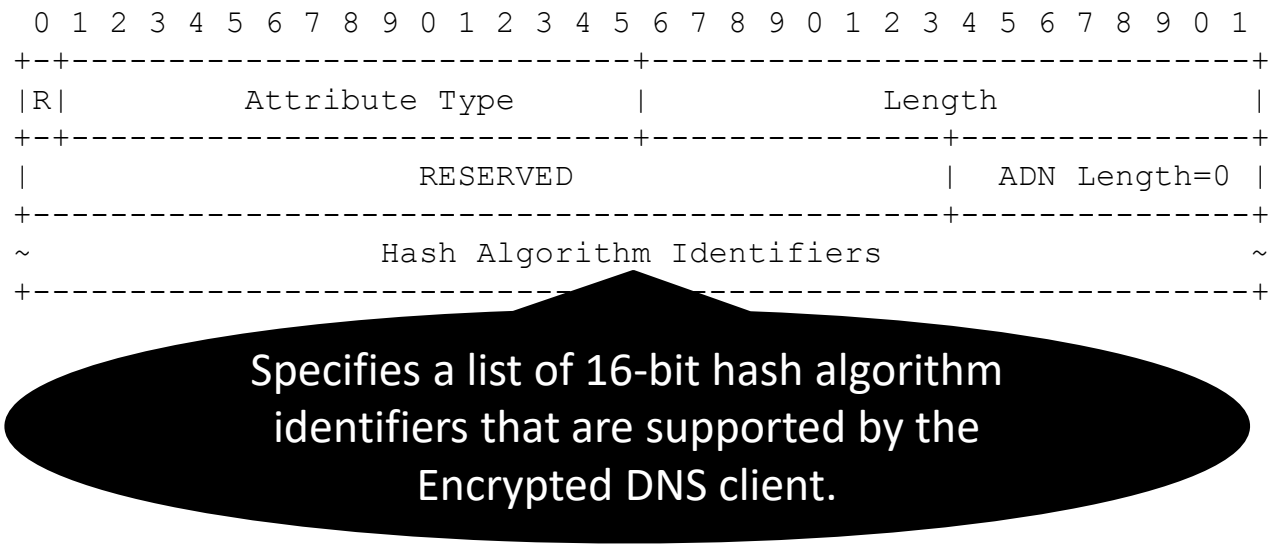
- ***Simplified design***
  - Align with recent I-D.ietf-add-dnr design (*ADD WG*)
  - Leverages SVCB

- Address comments raised by the WG in IETF#110
  - Move the deployment section to an appendix
  - ***Rely upon IKEv2 to validate the end-entity certificate***, instead of PKI: Add a new attribute ENCDNS_DIGEST_INFO

Simplified Design

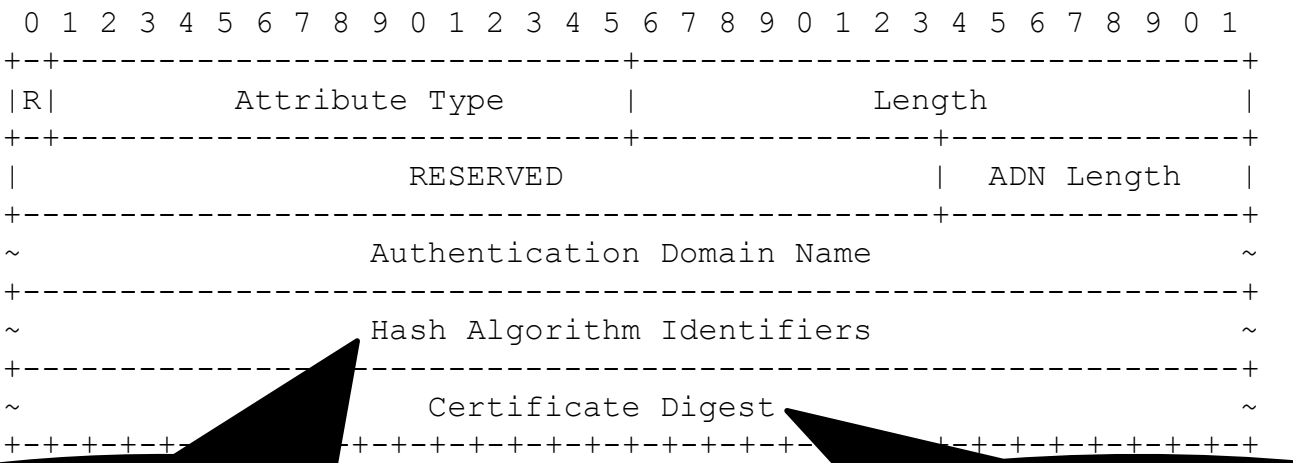


ENCDNS_DIGEST_INFO: Request



- No need for a new IANA registry
  - Values are taken from ***the IANA IKE's Hash Algorithm identifiers***
- SHA2-256 is mandatory-to-implement

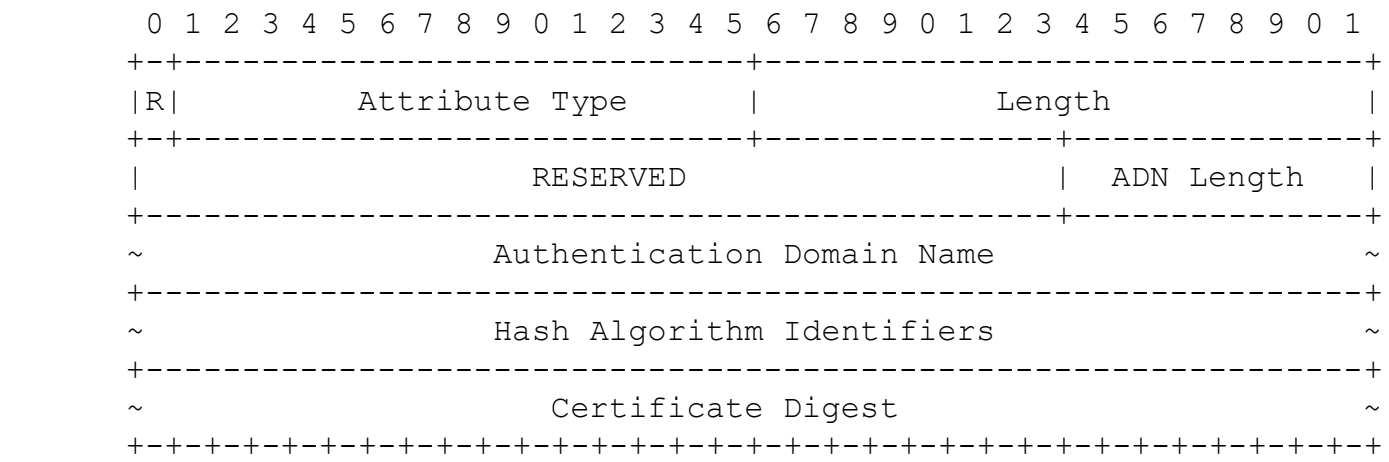
ENCDNS_DIGEST_INFO: Response

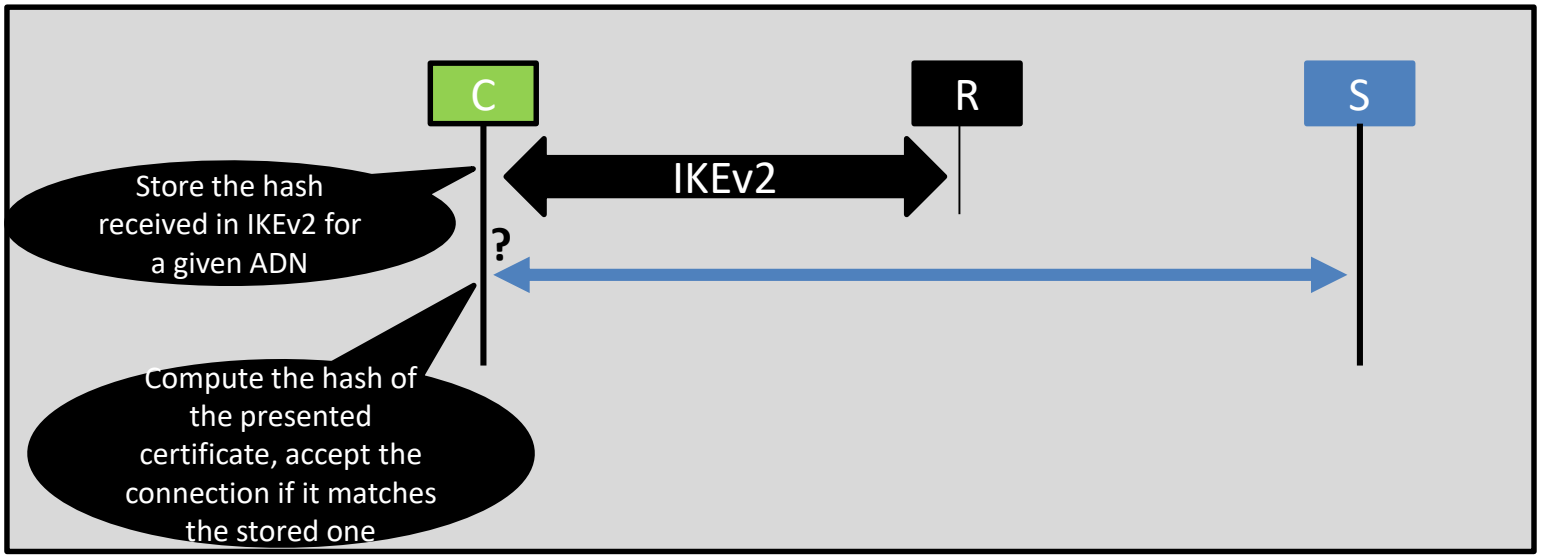


Specifies the hash algorithm identifier ***selected by the server*** to generate the digest of its certificate

Includes the digest of the Encrypted DNS server certificate using the selected hash algorithm

ENCDNS_DIGEST_INFO: Response





Next Steps

- Consider WG adoption

Presentations

- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- **Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov**
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- IKEv2 support for per-queue Child SAs – Paul Wouters

Large Payloads in IKEv2

draft-tjhai-ikev2-beyond-64k-limit-01

CJ Tjhai (Post-Quantum)
Tobias Heider (genua GmbH)
Valery Smyslov (ELVIS-PLUS)

IETF 111

Motivation

- draft-ietf-ipsecme-ikev2-multiple-ke addresses issues of using large keys for Key Exchange methods (common in PQC) in IKEv2
- This draft still limits the size of any single public key to 64K – the maximum size of IKEv2 payload
  - most NIST Third Round Candidate Algorithms fit into this restriction
  - notable exception - Classic McElice PQKE which smallest public key is 255 KB
- However, some national regulators (e.g. BSI) **recommend** using Classic McElice PQKE
- It is also anticipated that PQ Digital Signatures will be used in IKEv2
  - Some NIST Third Round Candidate Digital Signature Algorithms have either public key size (Rainbow) or signature size (Picnic) greater than 64 KB

Goals

- The goal of the document is to define a way for using some specific data blobs in IKEv2 if they grow beyond 64K
  - public keys for key exchange methods (KE)
  - signatures (AUTH)
  - certificates (CERT)
- The defined mechanism must be backward compatible
- Reliability of transferring large data in IKEv2 should be addressed
- Performance of IPsec traffic should not degrade
- The defined mechanism must be simple and must introduce minimal changes to IKEv2

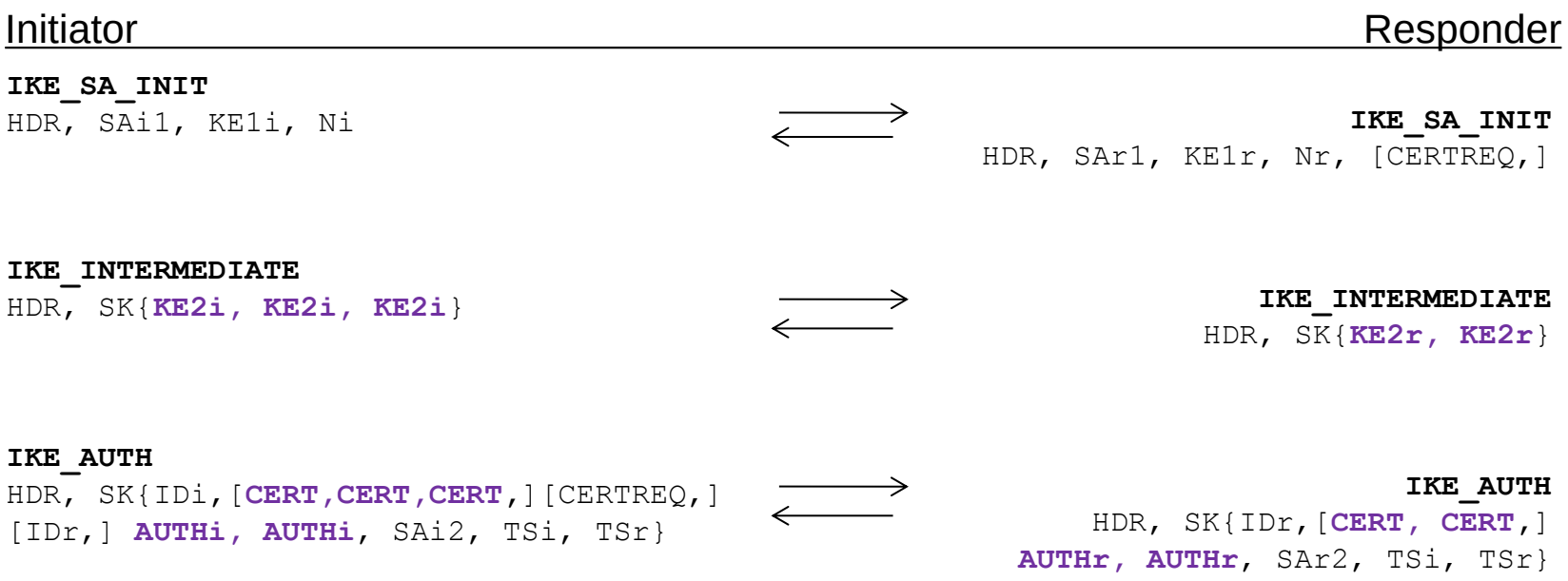
Not Goal

- There is no goal to define a generic mechanism for IKEv2 which would allow **any** payload be greater than 64K

Proposed Approach

- If amount of data doesn't fit into a single payload then split data into chunks less than 64K and put them into a sequence of payloads of the same type; receiving end will concatenate data from a sequence of payloads having the same type
  - this approach works well if only one payload of this type may appear in the message according to IKEv2 (true for KE and AUTH, not true for CERT, but can be worked around)
  - if such sequence of payloads appears inside Encrypted payload, then the Length field of the Encrypted payload would be overflowed, but this doesn't matter, since the length of Encrypted payload can always be deduced from the length of IKE message, so we can use value 0 for it

Example


Initiator
Responder
IKE_SA_INIT
HDR, SAi1, KE1i, Ni
IKE_SA_INIT
HDR, SAr1, KE1r, Nr, [CERTREQ,]
IKE_INTERMEDIATE
HDR, SK{KE2i, KE2i, KE2i}
IKE_INTERMEDIATE
HDR, SK{KE2r, KE2r}
IKE_AUTH
HDR, SK{IDi,[CERT,CERT,CERT,][CERTREQ,]
[IDr,] AUTHi, AUTHi, SAi2, TSi, TSr}
IKE_AUTH
HDR, SK{IDr,[CERT, CERT,]
AUTHr, AUTHr, SAr2, TSi, TSr}

Changes from -00 version

- IKE Fragmentation is now mandatory for both UDP and TCP transport
  - with TCP the size of a single IKE message is still limited to 64 KB, so we need IKE Fragmentation to transmit larger messages (with TCP they may be fragmented to 64 KB fragments)
- Mixed Transport Mode is introduced
  - with this mode IKE starts from UDP port 4500, then switches to TCP on the first INTERMEDIATE exchange and continues to use TCP for all subsequent exchanges; however, Child SAs created with this IKE SA use either direct transport or UDP encapsulation
  - this mode is negotiated by exchange of new notification IKE_OVER_TCP
  - Mixed Transport Mode allows IKE to reliably transfer large blobs of data still avoiding performance implications of using TCP for ESP
- Added clarifications for tweaking Length field of Encrypted Payload

Future Discussion

- DoS attacks are an important concern for this extension; we are going to discuss how to defend against them in next version

Thanks

- Comments? Questions?
- WG adoption?

Presentations

- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- **IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei**
- IKEv2 support for per-queue Child SAs – Paul Wouters

IKEv2 Optional SA&TS Payloads in Child Exchange

https://datatracker.ietf.org/doc/draft-kampati-ipsecme-ikev2-sa-ts-payloads-opt/

Sandeep Kampati (Huawei)

Wei Pan (Huawei)

Paul Wouters (Aiven)

Meduri Bharath (Mavenir)

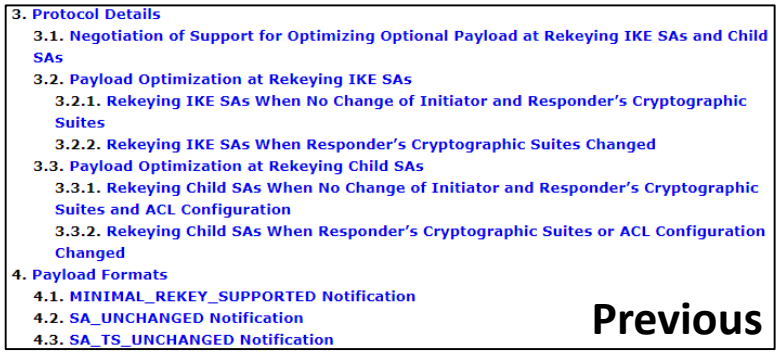
Meiling Chen (CMCC)

IETF 111, Online

July 2021

Updates from -02 to -07

- Whole solution is much simpler, text is more readable
  - 3 steps of optimization:
    - Negotiation of support for rekey optimization
    - Initiator and responder omit the SA payloads at rekeying IKE SAs
    - Initiator and responder omit the SA and TS payloads at rekeying Child SAs
  - No more consideration for the situation of configuration change
  - 2 new Notify Message type notifications are needed (Previous was 3)



- Two co-authors added: Paul Wouters (Aiven), Meiling Chen (CMCC)

3 Steps of Optimizations

- Negotiation of Support for OPTIMIZED REKEY

```
Initiator                           Responder
-----------------------------------------------------------------------
HDR, SK {IDi, [CERT,] [CERTREQ,]
    [IDr,] AUTH, SAi2, TSi, TSr,
    N(OPTIMIZED_REKEY_SUPPORTED)} -->
                                <-- HDR, SK {IDr, [CERT,] AUTH,
                                        SAr2, TSi, TSr,
                                        N(OPTIMIZED_REKEY_SUPPORTED)}
```

- Optimized Rekey of the IKE SA

```
Initiator                           Responder
-----------------------------------------------------------------------
HDR, SK {N(OPTIMIZED_REKEY),
    Ni, KEi} -->
                                <-- HDR, SK {N(OPTIMIZED_REKEY),
                                        Nr, KEr}
```

- Optimized Rekey of Child SAs

```
Initiator                           Responder
-----------------------------------------------------------------------
HDR, SK {N(REKEY_SA), N(OPTIMIZED_REKEY),
    Ni, [KEi,]} -->
                                <-- HDR, SK {N(OPTIMIZED_REKEY),
                                        Nr, [KEr,]}
```

Open questions

- 1) Should the SUPPORTED notify mean that peers MAY/SHOULD/MUST use this method?
- 2) Alternatively, the SUPPORTED notify could have a payload that signifies whether the old method is supported or not.
- 3) When a Child SA was negotiated with PFS, what should an optimized rekey do when there is no KE payload? Send INVALID_KE_PAYLOAD?

Next Steps

- Ask for WG adoption
- Discuss and close the open questions
- Looking for implementations to do interop testing

Presentations

- Hybrid IKEv2 Interoperability Testing – Valery Smyslov
- Improvements for post-quantum IKEv2 – Daniel Herzinger
- Management of IPTFS (Yang and SNMP draft) – Christian Hopps
- Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS – Valery Smyslov
- Beyond 64KB Limit of IKEv2 Payloads – Valery Smyslov
- IKEv2 Optional SA&TS Payloads in Child Exchange – William Panwei
- **IKEv2 support for per-queue Child SAs – Paul Wouters**

IKEV2 SUPPORT FOR PER-QUEUE CHILD SA

DRAFT-PWOUTERS-IPSECME-MULTI-SA-PERFORMANCE-00

IPsec, IETF 111

July 2021

Antony Antony, Tobia Brunner, Steffen Klassert, Paul Wouters

Goal of the draft:

- 40-100 Gbps wire speed IPsec using multiple CPU cores
- An unencrypted link of 10 Gbps or more is commonly reduced to 2-5 Gbps when IPsec is used to encrypt the link using AES-GCM.

- By using the implementation specified in this draft, aggregate throughput increased from 5Gbps using 1 CPU to 40-60 Gbps using 25-30 CPUs

Changes since last version

From draft-pwouters-multi-sa-performance
To draft-pwouters-ipsecme-multi-sa-performance

- Due to fixed name, diff not linked, see manual diff
- Separate info notifies for QoS and CPU case
- Clarified terms: Initial Child SA → Fallback Child SA
- Always require an INFO notify for an Additional Child SA
- Negotiate the maximum number (not minimum)
- Attempt to clarify QoS case
- Added some operational considerations
- Clarified case when not having per-queue ACQUIRE

Questions for WG: QoS

- Should we remove/split QoS in separate draft?
  - Authors have little QoS experience, no p-QoS code.
- Negotiate “all” QoS / flows at once? There is no variable number – the number is “all the different ones”
- Need to request ALL combinations? Or just ones you want (eg “bulk” and “voip”)
- How does IPv4 QoS & IPv6 flow label combine in TS?
- Do we need a new “reject this QoS/flow” TS_ error code?
- Can one combine per-CPU and per-QoS ? We don’t really know.

Questions for WG: Other

- When too many Child SA's, return TS_UNACCEPTABLE, ~~NO_ADDITIONAL_SAS~~, or a new error code?

- Considerable effort (two years) made by implementations (Linux, libreswan, strongswan).
  - Need to know if WG wants to move forward or not.

Open Discussion

- Other points of interest?