

IP Security Maintenance and Extensions (IPsecME) WG

IETF 117, Wednesday, July 28th, 2023

Chairs: Tero Kivinen
Yoav Nir

Responsible AD: Roman Danyliw

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

We need volunteers to be:

- Two note takers

MeetEcho: <https://meetings.conf.meetecho.com/ietf117/?group=ipsecme&short=&item=1>

Notes: <https://notes.ietf.org/notes-ietf-117-ipsecme>

Agenda

- Note Well, technical difficulties and agenda bashing – Chairs (5 min) (15:30-15:35)
- Document Status – Chairs (10 min) (15:35-15:45)
- Presentations
 - IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters (15 min) (15:45-16:00)
 - An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz (10 min) (16:00-16:10)
 - Traffic Selector for IKEv2 to add support DSCP – Daniel Migault (5 min) (16:10-16:15)
 - IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault (5 min) (16:15-16:20)
 - Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault (5 min) (16:20-16:25)
 - Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh (10 min) (16:25-16:35)
 - Use of Reliable Transport in the IKEv2 – Valery Smyslov (10 min) (16:35-16:45)
 - Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert (10 min) (16:45-16:55)
- Adoption calls – Chairs (5 min) (16:55-17:00)
- AOB + Open Mic (0 min)

WG Status Report

- Published as RFCs
 - Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2) [RFC9370](#)
 - Deprecation of the Internet Key Exchange Version 1 (IKEv1) Protocol and Obsoleted Algorithms [RFC9395](#)
- RFF Editor queue:
 - [draft-ietf-ipsecme-labeled-ipsec](#)
 - [draft-ietf-ipsecme-add-ike](#)
- Publication requested:
 - Currently none

WG Status Report

- Waiting for write-up / AD Followup:
 - [draft-ietf-ipsecme-g-ikev2](#)
- Working Group Last Call:
 - [draft-ietf-ipsecme-auth-announce](#)
- Work in progress:
 - [draft-ietf-ipsecme-ikev2-sa-ts-payloads-opt](#)
 - [draft-ietf-ipsecme-multi-sa-performance](#)

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

Presentations

- **IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters**
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert



IKEV2 OPTIMIZED REKEY SUPPORT

DRAFT-IETF-IKEV2-SA-TS-PAYLOADS-OPT

IPsec, IETF 117
July 2023

Paul Wouters

Remaining Issues: IPCOMP

- Further clarify if IPcomp used, rekey MUST contain an IPCOMP_SUPPORTED payload with CPI and same compression algorithm.
- RFC 7296 states:
 - This Notify message may be included only in a message containing an SA payload negotiating a Child SA
 - but we have no SA payload in an Optimized Rekey.
- Do these issues require an Updates: 7296 addition ?

Remaining Issues: Initial Child SA

- Initial Child SA protected under the IKE SA Key Exchange Method.
- Peers do not know if other end wants to use PFS for Child SA rekeys.
- And for some (unwise) implementations, allow a different Key Exchange type/strength for a rekeying child SA than the initial IKE SA Key Exchange

Remaining Issues: Initial Child SA

- If peers have PFS mismatch, OPTIMIZED_REKEY will fail.
- Should it sent INVALID_KEY ?
- Should it then retry OPTIMIZED_REKEY or go back to “classic” ?
- Solution 1: Require same KE type for IKE and Child SAs when using Optimized Rekeys
- Solution 2: Send Notify in Initial Exchange for child KE type
- Solution 3: Always do 1 “classic” rekey, remember the KE type, then subsequently use optimized rekeys

Remaining Issues: Critical Bit

- Draft states Critical Bit should be set for the new Notify payload – this is wrong
- Solution 1: Just remove it. Nothing else needed.
- Solution 2: Change OPTIMIZED_REKEY from a Notify payload to its own type of payload, then set critical bit on it.

Next steps ?

- Confirm consensus of previous slides answers on the list
- Push out new draft
- Start WGLC next week?

(please don't let this take another 4 months)

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- **An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz**
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation

Ke Xu, Jianping Wu, Yangfei Guo, Benjamin M. Schwartz, Haiyang Wang

draft-xu-ipsecme-risav: <https://datatracker.ietf.org/doc/draft-xu-ipsecme-risav/>
Github: <https://github.com/bemasc/risav/>

IETF 117

Jul. 2023



Site-to-site IPsec is awesome

- It defeats IP spoofing
 - Any packets arriving from the peer's IP range without IPsec can be dropped.
- It improves security
 - IPsec makes traffic tamper-resistant.
- It improves privacy
 - IPsec tunnels allow (but do not require) encryption.
- It is well-understood and technically mature.
 - With IKEv2, etc.
- It can be extremely fast.
 - 1 Tbps IPsec ESP demonstrated in pure software in 2021.
- Any pair of networks can use it.
 - There's no need for the participants to be direct BGP peers.

Site-to-site IPsec setup protocol (today)

From: alice@corp1.example
To: bob@corp2.example
Subject: Setting up a tunnel

Hi Bob, this is Alice is from Corp1. Would you be interested in setting up a site-to-site tunnel with us? We have an IKEv2 gateway running at 2001::db8:1 (see attached certificate hash).

From: bob@corp2.example
To: alice@corp1.example
Subject: Re: Setting up a tunnel

Hi Alice, that sounds great. We can get each other's IP ranges from the RPKI database. We'll authenticate IKEv2 with the attached client certificate, and use the tunnel for all traffic.



How do we get more site-to-site IPsec?

Automate!

1. Define a config format equivalent to Alice & Bob's emails.
2. Each participant publishes their config in the RPKI database.
3. All participants sync the RPKI database as usual.
4. Each participant connects to all the other participants.

Result: $O(N^2)$ IPsec associations with $O(N)$ human work.

This is the core idea of RISAV. Everything else is technical details, **subject to change**, to solve problems like:

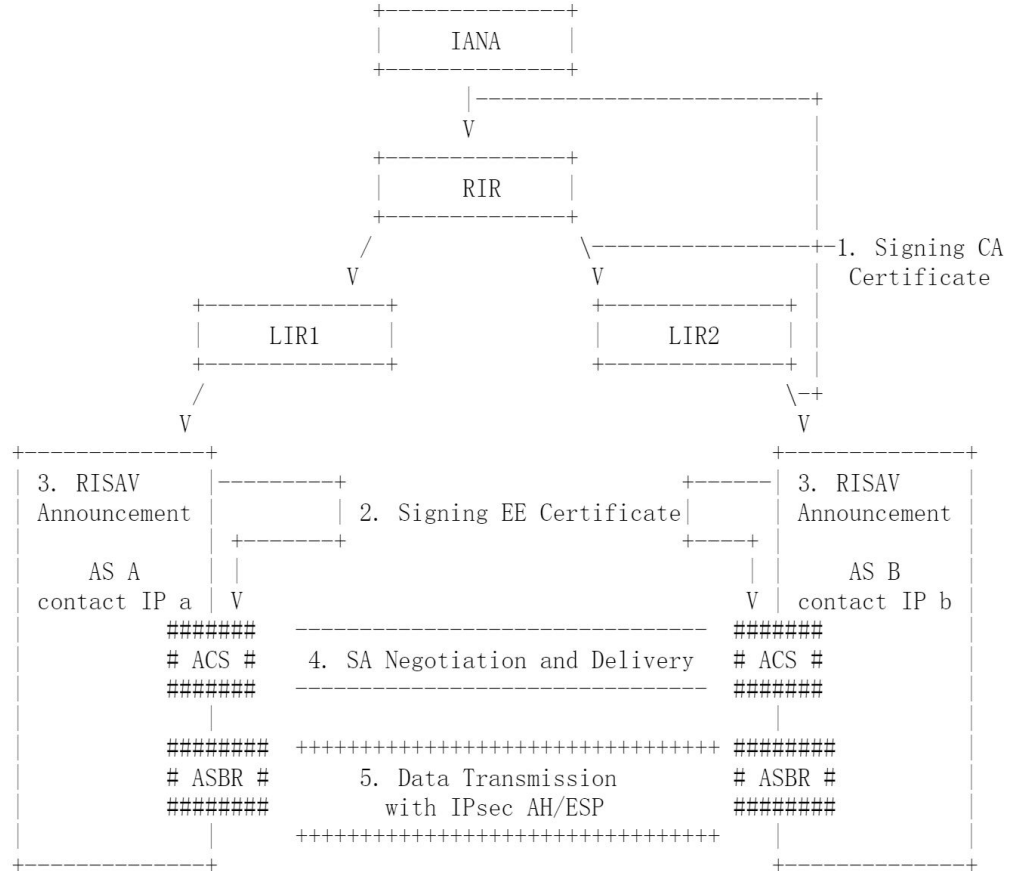
- *How do we make IPsec scale to serve networks with many gateways and sites without disrupting their routing?*
- *How do we minimize and tolerate MTU loss?*
- *Can make it easy and safe to turn the tunnel on and off?*

RISAV Overview

- cryptographically-based inter-AS SAV protocol
- RPKI + IPsec compatible
- add MAC at source ASBR and delete it at destination ASBR

*IP Source Address is viewed as correct **only if** the packet carries a correct MAC.*

RISAV does not require encrypting the whole packet or not aim to defend a specific attack. It just aims to provide SAV.





Network-to-Network Data Plane

Transport mode

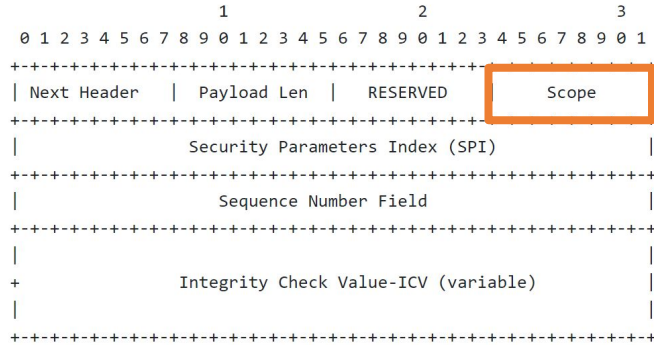


Figure 2: Updated AH Format.

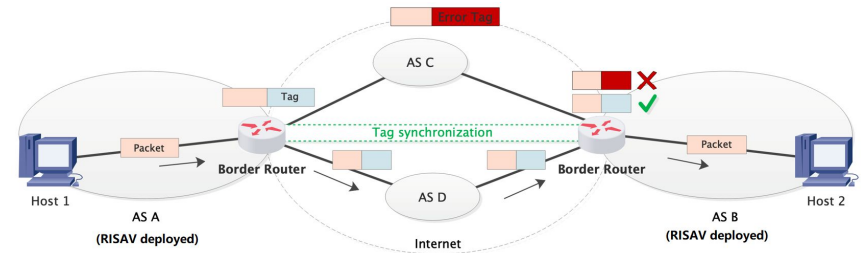
- ❖ ONLY the “Scope” field, which identifies the scope of protection for RISAV AH, is different from the original AH.
 - 0 for IP and 1 for AS; others not defined.
- ❖ Only used for AS-to-AS communication
- ❖ Only indexed by SPI and counterpart ASN regardless of src IP or dst IP in SAD
- ❖ Transparent to the end hosts.

Tunnel mode

- ❖ ESP encapsulation
- ❖ Tunnel is built with current ASBR and ACS’s contact IP of another AS
- ❖ ASBR maintains its own SAD indexed by SPI and counterpart ASN

RISAV implementations **MUST** support transport mode, and **MAY** support tunnel mode.

- ❖ USE_TRANSPORT_MODE notification





Closing remarks

- RISAV treats the Internet as a true “network of networks”
- RISAV provides clear benefits for participants even when only fractionally deployed.
 - e.g. if $x\%$ of networks have RISAV, joining RISAV reduces your amplification-reflection attack volume by $x\%$ (on average).
- The design has been getting simpler as other IPsec drafts propose solutions to key protocol scalability challenges.
- Seeking working group adoption
 - IPSECME, SIDRops, or elsewhere?
- Suggestions are welcomed.

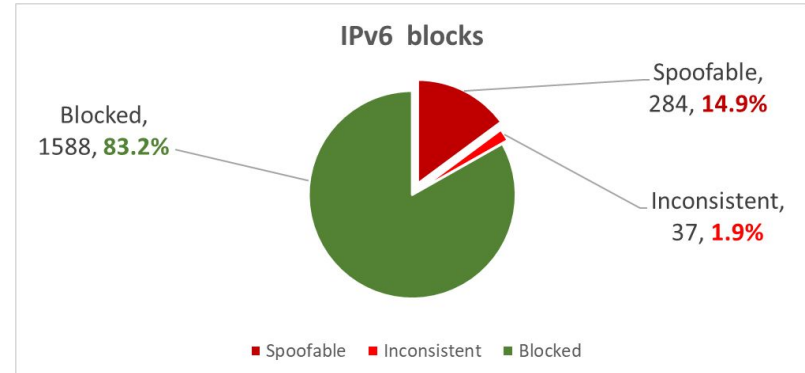
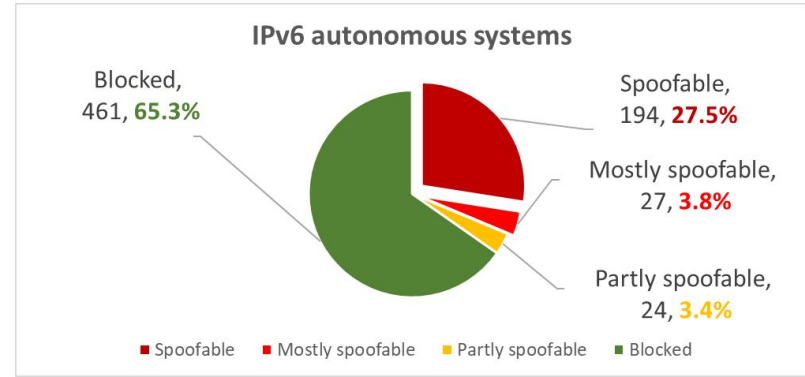
Thanks

2023/7/3

SAV Problem Statement

- **Vulnerability:** It is difficult to resist attacks by disabling the IP source address.
- **Traceability:** Attackers could conceal location and identity.
- **Manageability:** It is difficult to realize billing and other management through the IP source address.

This is not properly handled.





Control Plane

Enabling RISAV

- ❖ Announcing that this AS supports RISAV.
- ❖ Publishing contact IPs.
- ❖ RISAVAnnouncement: a Signed Object, testing for indicating the reliability of contact IP.
- ❖ Performing IPsec session initialization (i.e. IKEv2).

```
RISAVAnnouncement ::= SEQUENCE {  
    version [0] INTEGER DEFAULT 0,  
    asID      ASID,  
    contactIP SEQUENCE (SIZE(1..2)) OF IPAddressFamily,  
    testing   BOOLEAN DEFAULT FALSE }
```

Green Channel

- ❖ A channel established only between pair ACSes.
- ❖ For rebooting quickly and imperceptible
- ❖ When it enabled, ASBRs don't perform RISAV validation.

Disabling RISAV

- ❖ Targeted Shutdown
 - NO pair of inbound-outbound SAs. => strictly unidirectional SA.
 - If one AS sends NO_ADDITIONAL_SAS to its peer, it means the peer MUST halt all further RISAV negotiation temporarily.
 - Deleting all SAs and rejecting new ones.
- ❖ Total Shutdown
 - Apply a targeted shutdown
 - Stop requiring RISAV authentication of incoming packets.
 - Remove the "RISAVAnnouncement" from the RPKI Repository.
 - Wait at least 24 hours.
 - Shut down the contact IP.

MTU Handling and Replay Protection

Choose a **minimum** acceptable “**inner MTU**” and reject RISAV negotiations whose inner MTU is **lower than** inner MTU.

- Prior knowledge of the outer MTU
- Estimation of the outer MTU

ICMP PACKET TOO BIG(PTB)

- ❖ Transport Mode
 - MTU value reduced by the total length of RISAV AH header
- ❖ Tunnel Mode
 - Be treated as single IP hop
 - Oversize will cause generating PTB

MTU Estimation

- ❖ Initial estimation
 - PMTUD (RFC 7383)
- ❖ MTU monitoring

Traffic Selector and Replay Status

- ❖ Simplest RISAV Configuration
 - Single Child SA (**SHARING one**)
 - TSi lists all the IPs of sending AS and TSr lists all the IPs of receiving AS

Enabling Replay Protection

- ❖ Sender creates many Child SAs and narrow the TSi.
- ❖ each SA is processed by a single receiving ASBR
- ❖ Tunnel Mode: route each SA to a specific ASBR using IKEv2 Active Session Redirect.
- ❖ Transport Mode:

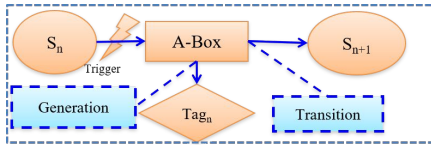
Disabling Replay Protection

- ❖ Set the REPLAY-STATUS indication to False in CREATE_CHILD_SA notification,
- ❖ and delete the SA if....

AS IP Ranges

Possible Extensions

Time-base Key Rotation



Time triggers the SM transit from $S(n)$ to $S(n+1)$ following the algorithm defined by two parties as well as generating the tags as the side product.

Header-only Authentication

It only authenticates the IP source address, IP destination address, etc.

An attacker could simply replace the payload, allowing it to issue an unlimited number of spoofed packets.

Static-static ECDH negotiation

Ideas from [RFC 6278](https://www.rfc-editor.org/rfc/6278)

It would allow ASes to agree on shared secrets simply by syncing the RPKI database.

Pros.

- Stateless

Cons.

- Novel IPsec negotiation mechanism



Others

Security Consideration

1. Threat model
 - a. Reply attack
 - b. Downgrade attack
2. Incremental benefit
3. Comparability
 - a. IPsec
 - b. Other SAVs

Operational Consideration

1. Reliability
2. Multiple ASBRs
3. Performance
4. NAT

Consistency with Existing Protocols

❖ IPv6

- MTU: minimum of 1280B.
{[MTU-Handling](#)}
- Header Modification: RISAV-AH
- IP address usage

❖ RPKI Usage

- RISAV fully falls squarely within the limits of usage of RPKI key material.

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- **Traffic Selector for IKEv2 to add support DSCP – Daniel Migault**
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

Traffic Selector for Internet Key Exchange version 2 to add support Differentiated Services Field Codepoints (DSCP)

`draft-mglt-ipsecme-ts-dscp`

Migault, Halpern, Parkholm, Liu

Goal

Ensuring that traffic associated with different QoS do take different SAs.

Specifying a new TS Type TS_DSCP for IKEv2 as an additional selectors for the SPD.

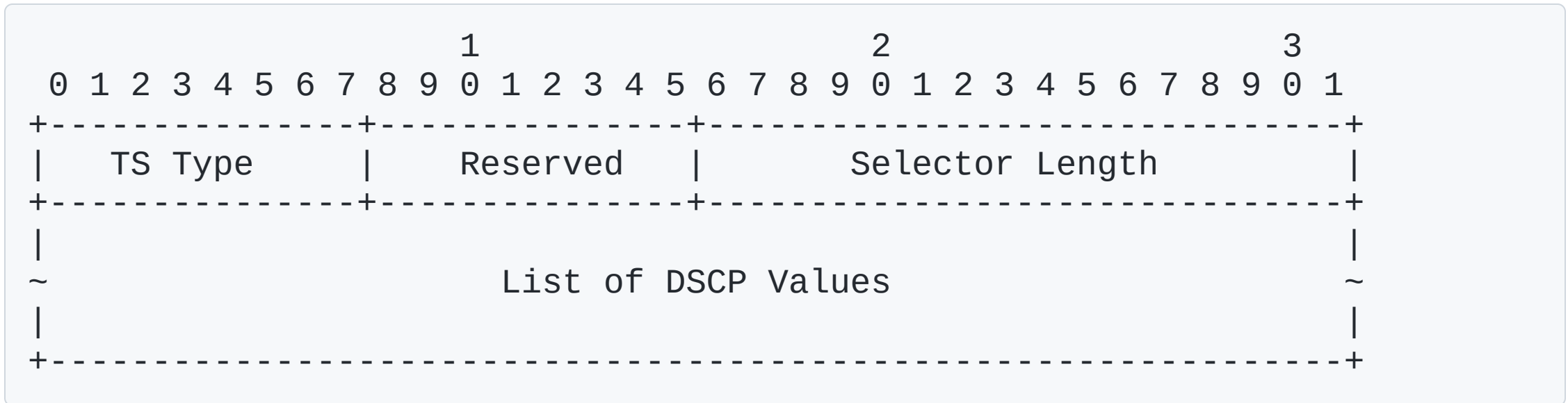
[RFC4301](#) Section 4.1 acknowledges that aggregating traffic with multiple DSCP over the same SA may result in inappropriate discarding of lower priority but recommends a **classifier** mechanism which dispatches the traffic over multiple SAs.

Such **classifier** results in inbound and outbound traffic may take SA negotiated via different IKEv2 sessions and thus makes:

- SA management more complex with an unnecessary SAs.

This is especially an issue with hardware implementations are designed with a limited number of SAs

Defining new TS that includes a range of acceptable DSCP: TS_DSCP_LIST



The CREATE_CHILD_SA request for rekeying a Child SA is:

```
Initiator                                Responder
-----
HDR, SK {N(REKEY_SA), SA, Ni, [KEi,]
  TSi, TSr}  -->
with:
  TSi = ( TS_IPV6_ADDR_RANGE, TS_DSCP )
  TSr = ( TS_IPV6_ADDR_RANGE )

                                     <-- HDR, SK {SA, Nr, [KEr,]
                                       TSi, TSr}

with:
  TSi = ( TS_IPV6_ADDR_RANGE, TS_DSCP )
  TSr = ( TS_IPV6_ADDR_RANGE )
```

Security

DSCP is a mutable field (not protected):

- only in Tunnel mode

DSCP is not a criteria used to define access control and we need to ensure that policies cannot be bypassed using different DSCP values.

- DSCP is always used in conjunction of TS_IP_ADDR_RANGE, so it is only a complementary information.
- We recommend all DSCP values be associated to the same rule, eventually DISCARD.

We are looking for a call for adoption.

Thanks!

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- **IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault**
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension

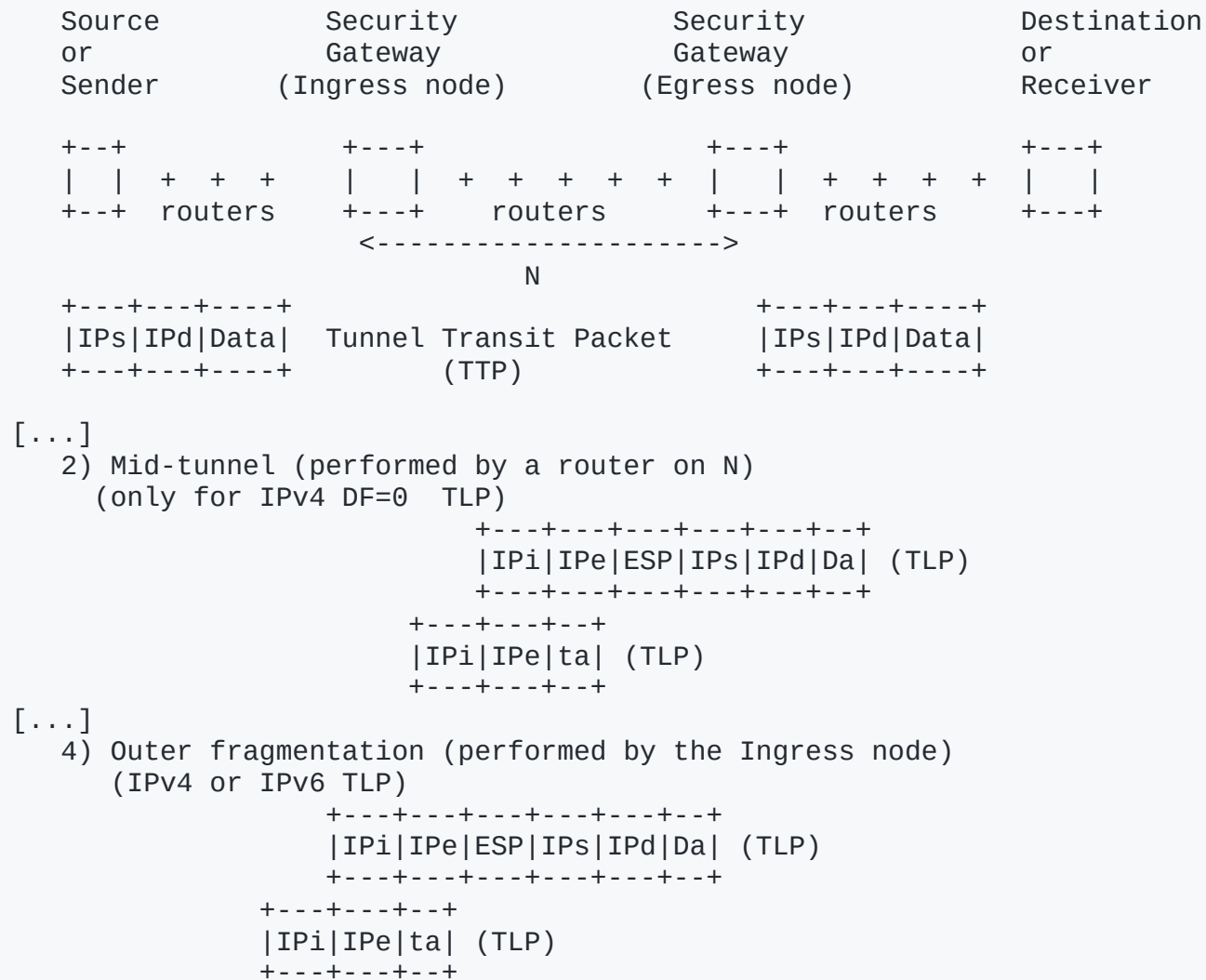
[draft-liu-ipsecme-ikev2-mtu-dect](#)

Liu, Zhang, Migault

Problem Statement

Fragments reassembling at the egress security gateway requires additional resources which under heavy load results in service degradations.

When Reassembly is observed ?



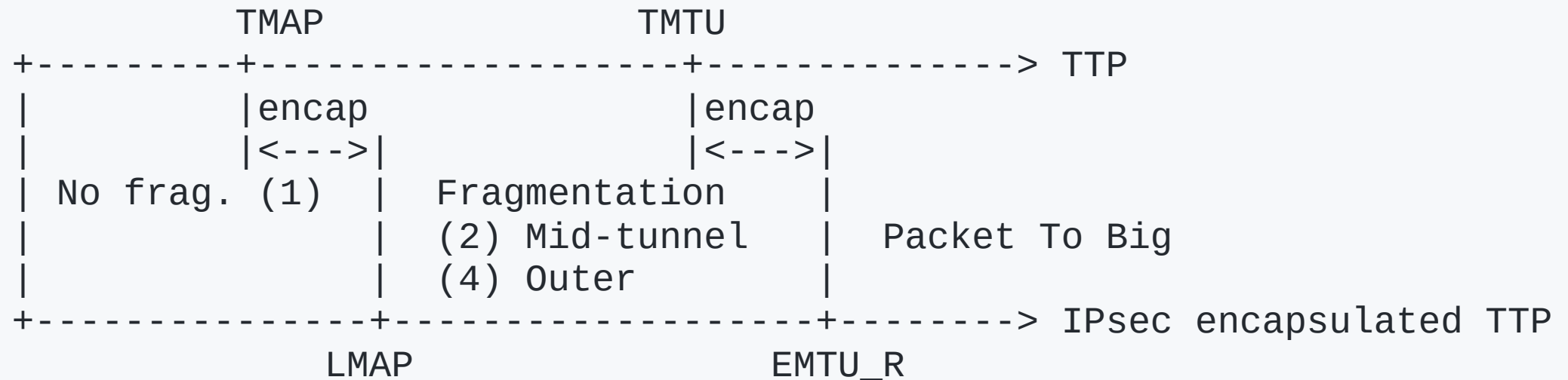
In both cases, the Egress nodes:

1. Reassembles fragments for an IPsec packet

- fragment < LMAP

2. Processes the reassembled IPsec packet

- (reassembled) IPsec encapsulated TTP < EMTU_R



We define two notification payload:

1. Link Maximum Atomic Packet Notification (LMTA)
 - To inform the Ingress node of the observed LMAP
2. Packet Too Big Notification (PTB)
 - to inform the Ingress node of the EMTU, LMTU

Given LMTA, EMTU_R the Ingress node is able to:

1. Compute the TMAP and TMTU
2. Inform the Source of appropriate TTP size (or perform inner fragmentation)

Illustrative Example (LMAP)

Source or Sender	Security Gateway (Ingress node)	Security Gateway (Egress node)	Destination or Receiver
------------------------	---------------------------------------	--------------------------------------	-------------------------------

```

+---+           +---+           +---+           +---+
| | + + + | | + + + + + | | + + + + | |
+---+ routers +---+ routers +---+ routers +---+

```

<----->
N

1) Mid-tunnel (performed by a router on N)
(only for IPv4 DF=0 TLP)

```

+---+---+---+---+---+---+
|IPi|IPe|ESP|IPs|IPd|Da| (TLP)
+---+---+---+---+---+---+
+---+---+---+
|IPi|IPe|ta| (TLP)
+---+---+---+

```

2) Egress node detects fragmentation

- a) it collects IPVersion the IP version of the first fragment as well as FragLen, the fragment length
- b1) If all segment can be reassembled reassemble and the reassembled packet properly decrypted a Link Maximum Atomic Packet Notification (LMAP) is sent.

is sent on the IKEV2 channel
[IKEV2]
<--- N(LMAP [IPVersion, FragLen])

3) Upon receiving the LMAP or optionally the ingress node

- a) Update the TMTU so that the Source performs source fragmentation with TTP packet that are not fragmented.

Source fragmentation
(IPv6 or IPv4)

```

+---+---+---+
|IPs|IPd|Da| (TTP)
+---+---+---+

```

```

+---+---+---+
|IPs|IPd|ta|
+---+---+---+

```

Where we are:

Remaining discussion:

- `ietf-intarea-tunnels` considers the router component - carrying the TTP - and the interface component - handling LTP - independent. Such independence between the Tunnel MTU (for TTP) and link layer MTU for (LTP) is provided by performing outer fragmentation when needed.
- RFC4301 considers the router component can adapt to the specific needs of the interface component. This is what we do here.

We follow RFC4301 and we are looking for adoption

Thanks.

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- **Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault**
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

ESP Header Compression Profile

`draft-mglt-ipsecme-diet-esp`

`draft-mglt-ipsecme-ikev2-diet-esp-extension`

Migault, Guggemos, Bormann, Schinazi

ESP Header Compression Profile (EHCP) defines a profile to compress communications protected with IPsec/ESP.

Compression / Decompression is based on the Generic Framework for Static Context Header (SCHC) [RFC8724].

- joint work with the SCHC WG

EHC Context	Possible Values	Reference	C / D
alignment	"8 bit", "32 bit"	ThisRFC	CT E
ipsec_mode	"Tunnel", "Transport"	RFC4301	CT E
tunnel_ip	IPv4, IPv6 address	RFC4301	CT E
esp_spi	ESP SPI	RFC4301	EE
esp_spi_lsb	0, 1, 2, 3, 4*	ThisRFC	EE
esp_sn	ESP Sequence Number	RFC4301	EE
esp_sn_lsb	0, 1, 2, 3, 4*	ThisRFC	EE
esp_encr	ESP Encryption Algorithm	RFC4301	CT E
ts_flow_label	True, False	ThisRFC	CT E
ts_ip_version	4, 6	ThisRFC	CT E
ts_ip_src_start	IP4 or IPv6 address	ThisRFC	CT E
ts_ip_src_end	IP4 or IPv6 address	ThisRFC	CT E
ts_ip_dst_start	IPv4 or IPv6 address	ThisRFC	CT E
ts_ip_dst_end	IPv4 or IPv6 address	ThisRFC	CT E
ts_proto_list	TCP, UDP, ..., 0	ThisRFC	CT E
ts_port_src_start	Port number	ThisRFC	CT E
ts_port_src_end	Port number	ThisRFC	CT E
ts_port_dst_start	Port number	ThisRFC	CT E
ts_port_dst_end	Port number	ThisRFC	CT E
ts_dsp_list	DSCP number	RFCYYYY	CT E

The following parameters need to be agreed:

- alignment
- esp_spi_lsb
- esp_sn_lsb
- ts_flow_label

draft-mglt-ipsecme-ikev2-diet-esp-extension defines an IKEv2 extension EHC_SUPPORTED to agree on these parameters.

Initiator

Responder

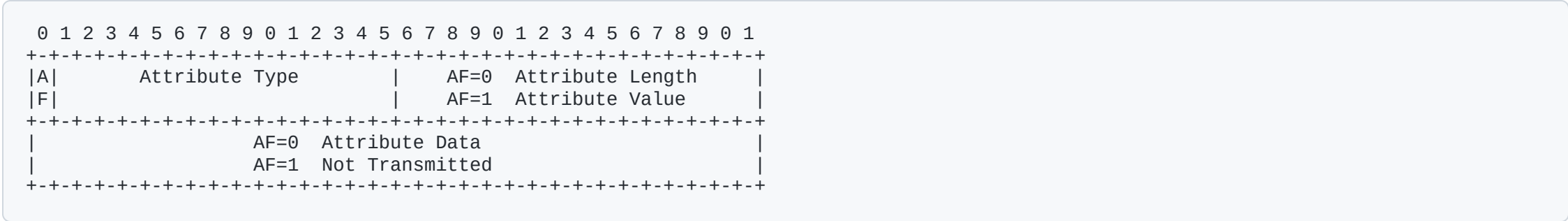
HDR, SA, KEi, Ni -->

<-- HDR, SA, KEr, Nr

HDR, SK {IDi, AUTH,
SA, TSi, TSr,
N(EHC_SUPPORTED
Proposal_1
param_a
...
param_i
...
Proposal_n
param_a
...
param_j)

<-- HDR, SK {IDr, AUTH,
SA, TSi, TSr,
N(EHC_SUPPORTED
ehc_context_id = "Diet-ESP"
selected_param_a
...
selected_param_m)

Parameters follow the same format as the Transform Attribute with AF=1



Parameter Code Point	Designation	Reference
0	ehc_context_id	ThisRFC
1	alignment	ThisRFC
2	esp_spi_lsb	ThisRFC
3	esp_sn_lsb	ThisRFC
4	ts_flow_label	ThisRFC
0 - 2 ** 15 - 1	unallocated	

EHC Context Identifier Value	Designation	EHC Context Reference
0	Diet-ESP	draft-mglt-ipsecme-diet-esp
1 - 2 ** 16 - 1	unallocated	

Where we are:

- We already implemented (contiki - without SCHC) Diet ESP:
 - [Diet-ESP: IP layer security for IoT](#) Journal of Computer Security, vol. 25, no. 2, pp. 173-203, 2017
- SCHC WG is integrating Diet-ESP in [openschc](#)

We believe the document is ready for adoption and we expect the implementation to be completed by the next IETF.

Thanks!

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- **Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh**
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

Anti-replay sequence number subspaces

draft-ponchon-ipsecme-anti-replay-subspaces

Mohsin Shaikh (presenter), Paul Ponchon, Hadi Dernaika,
Pierre Pfister, Guillaume Solignac
IETF 117 @ San Francisco

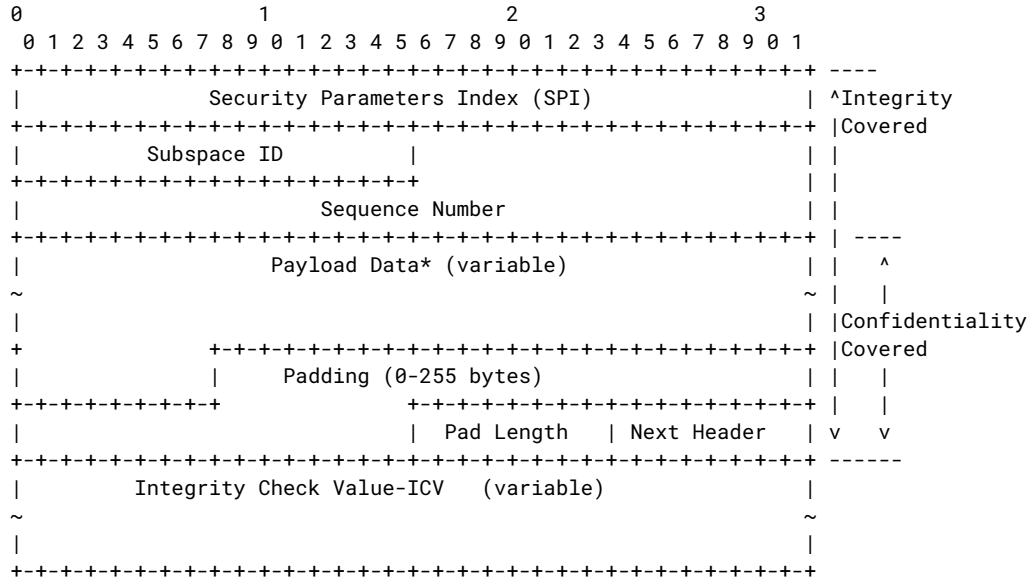
Quick Recap

- Proposal to use multiple sequence number subspaces as an alternative to creating multiple child SAs for multi-core performance
- Additionally support QoS and traffic engineered paths
- In draft-ponchon-ipsecme-anti-replay-subspaces-01 we added IKE transform to negotiate max number of subspaces
- We also increased the sequence number field in ESP packet to 64-bits with subspace ID stored in most significant 16 bits

Implementation

- We are working on implementing this in VPP open-source data-plane (<https://fd.io>)
- We are also working on a closed source implementation intended for deployment in Cisco Meraki's devices in the coming months

Updates in draft-02 since IETF 116



- The subspace ID must be between 0 to “N-1” where “N” is the max number of subspaces negotiated by IKE.
- ESN negotiation must be disabled as negotiating subspaces implies a 16-bit subspace ID and 48-bit sequence number counter.

Updates in draft-02 since IETF 116 (contd.)

- The 48-bit sequence number counter must not be allowed to cycle. A 1 Tbps would exhaust a subspace in over 938 hours (39 days). Assuming ethernet frames of 1500 bytes, $T = 2^{48} \text{ (pkts)} * 1500 \text{ (B/pkt)} * 8 / 10^{12} \text{ (bps)} = \sim 3.4 * 10^6 \text{ seconds} = \sim 938 \text{ hours}$.
- ICV calculation begins from start of ESP SPI field to end of ESP payload.
- IPR disclosure from Cisco Systems

- We support draft-mrossberg-ipsecme-multiple-sequence-counters being presented today which discusses using multiple sequence number counters
- We also welcome any support on the mailing list or in person
- We intend to deploy this on Cisco Meraki's ~2 million (and growing) devices worldwide
- Should the working group work to adopt and help us deploy a standard or should this remain a draft based extension?

Q&A

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- **Use of Reliable Transport in the IKEv2 – Valery Smyslov**
- Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert

Reliable Transport for IKEv2

`draft-smyslov-ipsecme-ikev2-reliable-transport`

Valery Smyslov
svan@elvis.ru

IETF 117

IKE Transport

- IKE originally used only UDP as a transport protocol
 - depending on the presence of NATs ESP either run over IP or were UDP encapsulated
- RFC 9329 extends IKEv2 to use TCP when UDP is unavailable
 - in this case ESP is always TCP encapsulated

Problem

- When post-quantum algorithms are employed (e.g. as defined in RFC 9370) size of IKE messages increases
 - use of IKE fragmentation eliminates problems with IP fragmentation, but doesn't address issues with possible congestion
 - simple retransmission mechanism of IKEv2 always resends the whole message even if only one fragment lost
- Using TCP solves the problem with transmission of large IKE messages, but
 - as ESP in this case also uses TCP, performance suffers (see Section 9 of RFC 9329)

Proposed Solution

- De-couple IKE and ESP transports (first proposed in draft-tjhai-ikev2-beyond-64k-limit)
 - use reliable transport (TCP) for IKE
 - use unreliable transport (IP, UDP) for ESP

Negotiation

- IKE_SA_INIT is initiated on UDP:4500 (to verify that UDP packets are routable)

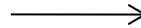
Initiator

Responder

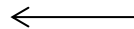
IKE_SA_INIT

HDR, SAi1, KEi, Ni,
[N(NAT_DETECTION_SOURCE_IP),
N(NAT_DETECTION_DESTINATION_IP),]
N(RELIABLE_IKE_TRANSPORT)

UDP



UDP

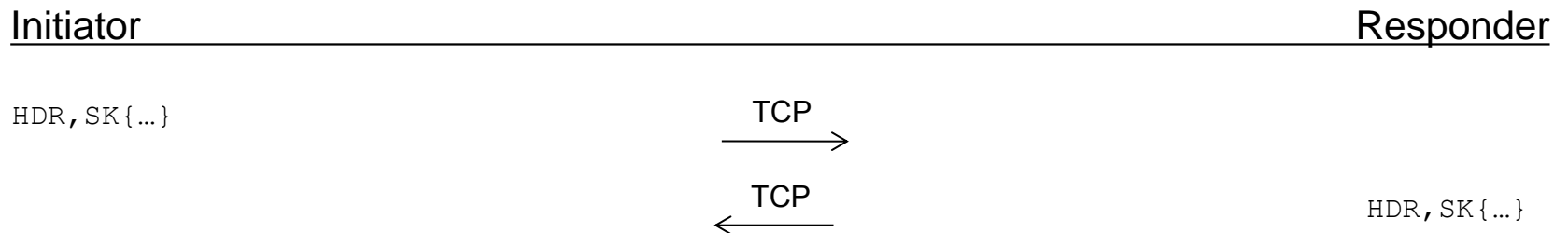


IKE_SA_INIT

HDR, SAR1, KEr, Nr,
[N(NAT_DETECTION_SOURCE_IP),
N(NAT_DETECTION_DESTINATION_IP),]
N(RELIABLE_IKE_TRANSPORT)

Use

- Once extension is negotiated, peers switch to TCP in the next exchange (IKE_INTERMEDIATE or IKE_AUTH) and continue to use TCP for all subsequent exchanges, including those of rekeyed IKE SAs



- ESP SAs are being created over IP or with UDP encapsulation

Some Details

- NAT keep-alive packets must still be sent over UDP if NAT is present
- Peers should not try to switch IKE SA to UDP if IP addresses changed and MOBIKE is in use
 - issue: no return routability check for UDP packets if IP addresses changed; perhaps INFORMATIONAL over UDP may be performed in this case
- Perhaps make reliable IKE transport negotiable (e.g. QUIC)?

Thank you!

- Comments?
- Questions?
- Any interest in this work?

Presentations

- IKEv2 Optional SA&TS Payloads in Child Exchange – Paul Wouters
- An RPKI and IPsec-based AS-to-AS Approach for Source Address Validation – Ben Schwartz
- Traffic Selector for IKEv2 to add support DSCP – Daniel Migault
- IKEv2 Link Maximum Atomic Packet and Packet Too Big Notification Extension – Daniel Migault
- Diet ESP: ESP Header compression, IKEv2 EHC – Daniel Migault
- Anti-replay sequence number subspaces for traffic-engineered paths and multi-core processing – Mohsin Shaikh
- Use of Reliable Transport in the IKEv2 – Valery Smyslov
- **Problem statements and uses cases for lightweight Child Security Associations – Steffen Klassert**

ESP Problem Statement

draft-mrossberg-ipsecme-multiple-sequence-counters-00

Steffen Klassert

ESP problems in today's networks

- Replay protection and packet reordering
 - Lot of proposals to fix this
 - Discussed in:
draft-mrossberg-ipsecme-multiple-sequence-counters-00
- Header and Trailer format
 - Might not fit anymore to all today's usecases
 - Still TODO in the draft

Replay protection and packet reordering

Problematic scenarios

- Multicore Software Processing
 - Needs synchronization between CPUs
 - Introduces reorder
- QoS support
 - Introduces reorder (intentional)
- Multipath
 - Introduces reorder
- Multicast
 - Needs synchronization between multiple senders
 - Introduces reorder

Possible solutions

1. Disabling replay protection
2. Increase anti-replay window sizes
3. Multiple IKE SAs
4. Multiple child SAs
5. sub-child SAs

Disabling replay protection

■ Advantages:

- solves all the reordering and synchronization issues
- No change in the standards needed

■ Disadvantages:

- Significantly lowers the level of security
- Receive side can't parallelize over multiple CPU cores

Disabling replay protection

- Solves:
 - Multicast
 - Multipath
 - QoS
- Does not solve:
 - Multicore
 - Solves just the transmit side problems
- But: Significantly lowers the level of security!

» **Not a solution for the general case!**

Increase anti-replay window size

■ Advantages:

- No change in the standards needed

■ Disadvantages:

- Needs synchronization (CPU cores / multicast senders)
- Receive side can't parallelize over multiple CPU cores
- Anti-replay windows can't grow indefinitely large
- Complex configuration (which size fits for a given usecase?)

Increase anti-replay window size

- Solves:
 - (Multipath)
 - (QoS)
- Does not solve:
 - Multicore
 - Multicast

» **Not a solution for the general case!**

Multiple IKE SAs

■ Advantages:

- No changes to existing standards required
- Independent sequence numbers and anti-replay windows
- Distinct SPIs allow RX side RSS or explicit steering

■ Disadvantages:

- Negotiation (time and communication) overhead
- state/memory overhead
- Unspecified failure model if a subset of SAs can't be established

Multiple IKE SAs

- Solves:
 - Multicore
 - Multicast
 - Multipath
 - QoS
- Does not solve:
 - Combinations of different cases (e.g. Multicore + QoS)
- But: Has scalability issues

» **Not a solution for the general case!**

Multiple child SAs

■ Advantages:

- Just minor changes to existing standards required
- Independent sequence numbers and anti-replay windows
- Distinct SPIs allow RX side RSS or explicit steering

■ Disadvantages:

- Negotiation (time and communication) overhead
 - Much less than ,multiple IKE SAs'
- state/memory overhead
 - Much less than ,multiple IKE SAs'
- Unspecified failure model if a subset of SAs can't be established

Multiple child SAs

- Solves:
 - Multicore → (draft-ietf-ipsecme-multi-sa-performance-01)
 - Multicast
 - Multipath
 - QoS
- Does not solve:
 - Combinations of different cases (e.g. Multicore + QoS)
- But: Still does not solve the generic case

» **Low hanging fruit, can solve a lot of usecases!**

Multiple sub-child SAs

- Idea: Use multiple sequence counters per child SA
- Encode the sequence number counter ID on some header field
- Multiple possibilities to achieve this
 - Not (yet) clear which one is the best

Multiple sub-child SAs

- Use bits of the sequence number
 - Draft-ponchon-ipsecme-anti-replay-subspaces-00
 - Reduces available sequence numbers per sub-child SA (ESN only)
 - Limited RSS support in current NICs
- Use some bits of the SPI
 - RSS support in current NICs
- Use a new header field
 - Draft-ponchon-ipsecme-anti-replay-subspaces-02
 - Transmit 64 bit sequence number
 - Biggest change to current standards
 - Limited RSS support in current NICs

Multiple sub-child SAs

■ Advantages:

- Independent sequence numbers and anti-replay windows
- Distinct SPIs allow RX side RSS or explicit steering
- No rekeying overhead
- Clean failure model

■ Disadvantages:

- Most complex change
- Needs care to avoid security implications
 - IVs must not repeat for counter modes
 - Rekey, byte, packet limits must apply to all sub-SAs combined

Multiple sub-child SAs

- Solves:
 - Multicore
 - Multicast
 - Multipath
 - QoS
 - Combinations of different cases (e.g. Multicore + QoS)
- But: Most complex change, has security implications
- Need to choose a header field: SPI, sequence number, new field

» **Might solve the general case!**

Header and Trailer format

Header and Trailer format

- Still TODO in the draft
- Need to identify usecases for different formats:
 - Google published PSP for HW offloads
 - Some discussion at IETF 108

Is the WG interested to continue this work?

WG Adoption calls

- draft-mglt-ipsecme-ts-dscp
- draft-liu-ipsecme-ikev2-mtu-dect
- draft-mglt-ipsecme-diet-esp
- draft-mglt-ipsecme-ikev2-diet-esp-extension
- draft-smyslov-ipsecme-ikev2-qr-alt
- draft-smyslov-ipsecme-ikev2-cookie-revised

Open Discussion

- Other points of interest?