Enhanced ESP
IETF 122

draft-klassert-ipsecme-eesp-02

Steffen Klassert, Antony Antony, Chris Hopps

EESP presentation at IETF 121

- Present packet format
- Discussed header fields
- Highlight changes to ESP

EESP presentation at IETF 121

- Present packet format
- Discussed header fields
- Highlight changes to ESP

This time: Requirements / Design / Why we need a new protocol

Why a new security protocol?

- ESP lacks flexibility and extensibility for modern needs
  - No version number in packet
  - New features must be negotiated
  - Not transparent to the network
- ESP is not designed for required use cases
  - High speed networks (1Tbit/s and more)
  - Efficient HW offloads
  - QoS
  - Multicore CPUs
  - SDN / Telemetry
  - .....

Requirements

EESP Requirements

- Performant in HW and SW
- Parallelizable (HW + SW)
- No additional SA negotiation
- Fast lookups
- Reduce fast memory usage (HW)
- Support replay protection
- QoS with minimal overhead
- Enable SDN / Telemetry
- Must coexist with ESP
- Future proof (extensible)

Design

Performance I

Sub Child SAs

- Parallelizable with replay protection
  - Separate sub Child SA for:
    - Each CPU core
    - Esch QoS class
- No extra negotiation needed
  - Keys are generated by a KDF
  - Keys can be generated ‘on the fly’ by HW
    - No need for additional fast memory (PSP requirement)

» **Parallelization with minimal overhead!**

Performance II

Packet Format

- No Trailer
  - Transport Mode:
    - Trailer info placed in front of the L4 payload
  - Tunnel Mode
    - Info can be derived from the IPv4/IPv6 header
- Only AEAD algorithms
  - Can use implicit IV (on the seq. nr.)
- Full 64 bits sequence number in the header
  - No implicit parts needed

» **Easy to parse the packet!**

Future proof

Flexibility / Extensibility

- 4 bits Version Number
- Variable Header Options possible
    - IPv6 extension header style

EESP Requirements (again)

- Performant in HW and SW ✓
- Parallelizable (HW + SW) ✓
- No additional SA negotiation ✓
- Fast lookups ✓
- Reduce fast memory usage (HW) ✓
- Support replay protection ✓
- QoS with minimal overhead ✓
- Enable SDN / Telemetry ✓
- Must coexist with ESP (IKEv2 ✓)
- Future proof (extensible) ✓

Feedback from the Nedev Conference 2025

March 10th – 14th

- IPv6 options TLVs are too generic
- Restrict TLV usage (like geneve)
- Find common protocol base
    - Split draft in base protocol an followups (data center etc.)
- Need to use unique HW timestamps as sequence number (and IV)

Next Steps

Focus on basic requirements

- Add a requirements section to the draft
- Make it simple
- Move stuff to followup drafts
- Decide for one variant of features

Major changes to ESP

- Adds Version Number (no need for new IP protocol numbers)
- Session ID
- Variable Header-Options possible
- Transmit 64 bit SeqNo (always use ESN)
- Use AEAD only
- Two packet formats (full/optimized)
- Remove implicit header parts
- Remove the trailer
- Remove TFC padding, use IPTFS instead

Why not use existing feature XYZ?

Why not use existing feature XYZ?

Not tailored to the problem!

Existing ‘solutions’ not accepted by users (20+ years)!

- Instead: Lots of proprietary solutions!
- People moved away from IPsec
  - Quick *connect (IETF masque WG)
  - Google PSP (**not standardized**)
  - Wireguard (**not standardized**)
  - OpenVPN (**not standardized**)
  - New Protocol by Ultra Ethernet Consortium (**not standardized)**

Need a **standardized** protocol that fits the requirements!

Make EESP a WG Adoption Candidate?