ESP Header Compression

draft-mglt-ipsecme-diet-esp

Migault - Guggemos - Bormann - Schinazi

EHC: flexible framework to compress ESP

Principle:

- EHC Strategy defines the orchestration of EHC Rules
- EHC Rules, EHC Context defines compression / decompression

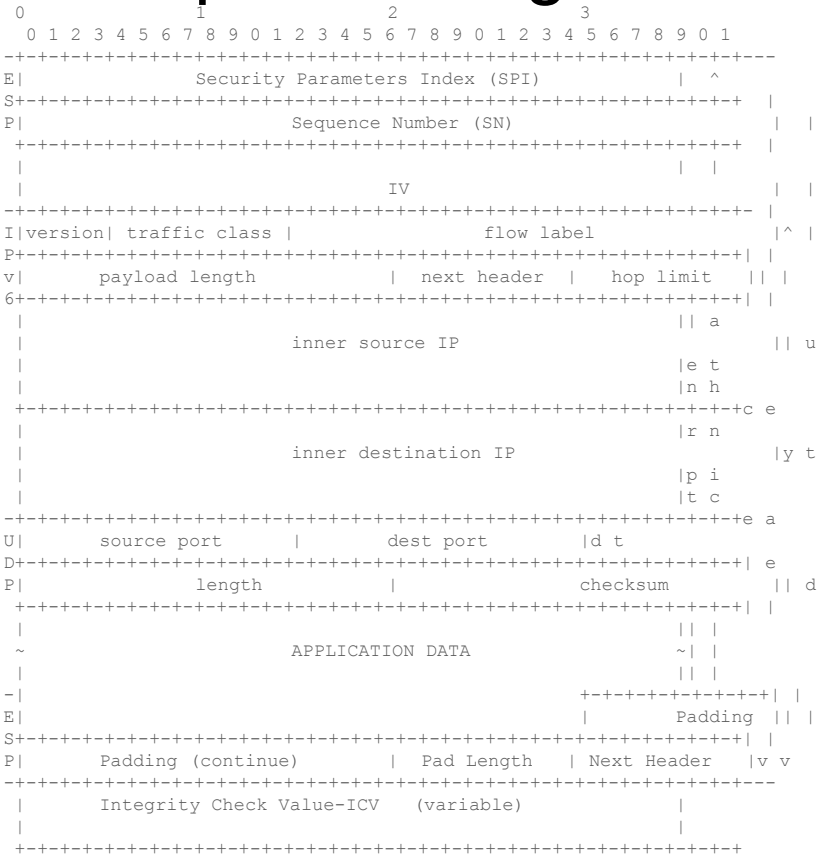
The document defines:

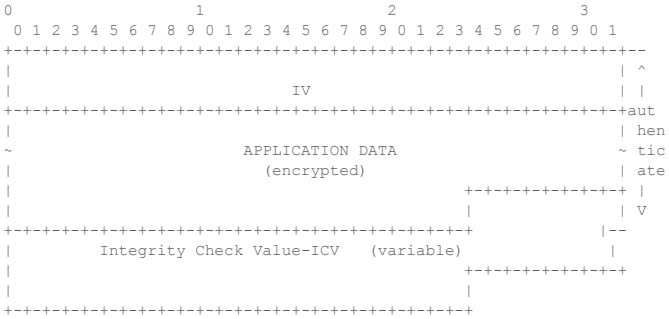
- EHC Rules, EHC Context
- Diet-ESP Strategy, Diet-ESP Context
    - Derive EHC Context for each EHC Rule
    - EHC Rules to apply

Example 1: Single UDP Session

EHC Rule	Context Attribute	Value
ESP_SPI	esp_spi_lsb	0
ESP_SN	esp_sn_lsb	0
	esp_sn_gen	
ESP_NH		
ESP_PAD	esp_align	8
IP6_OUTER	ip6_tcfl_comp	
	ip6_hl_comp	
IP6_LENGTH		
IP6_NH	l4_proto	in SA
IP6_HL_OUTER		
IP6_SRC	ip6_src	in SA
IP6_DST	ip6_dst	in SA
UDP_SRC	l4_source	in SA
UDP_DST	l4_dest	in SA
UDP_LENGTH		
UDP_CHECK		

Example 1 : Single UDP Session (61 bytes)

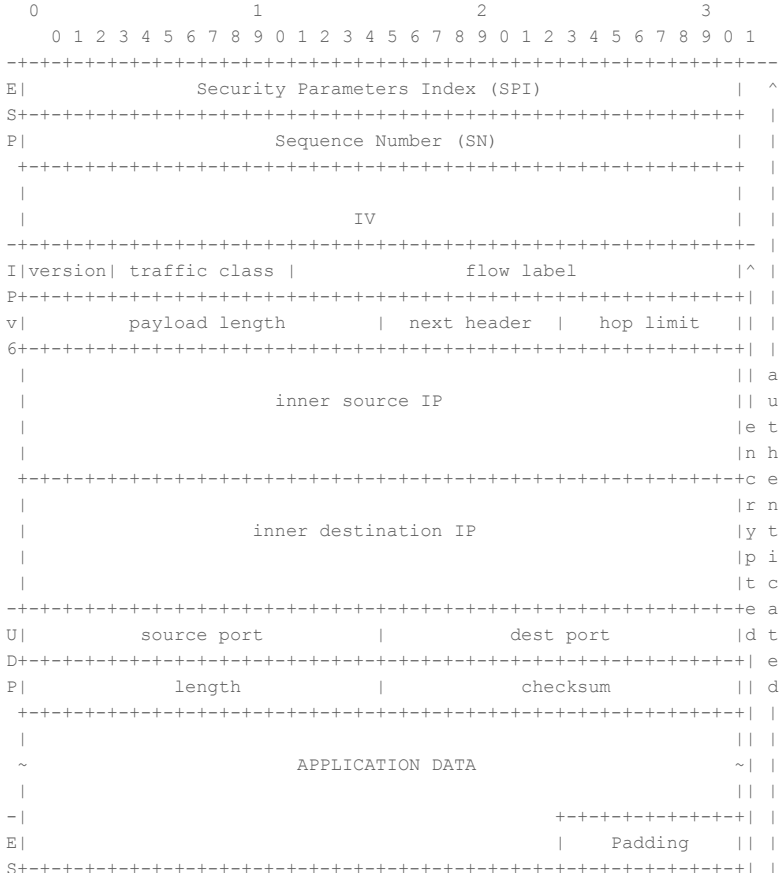


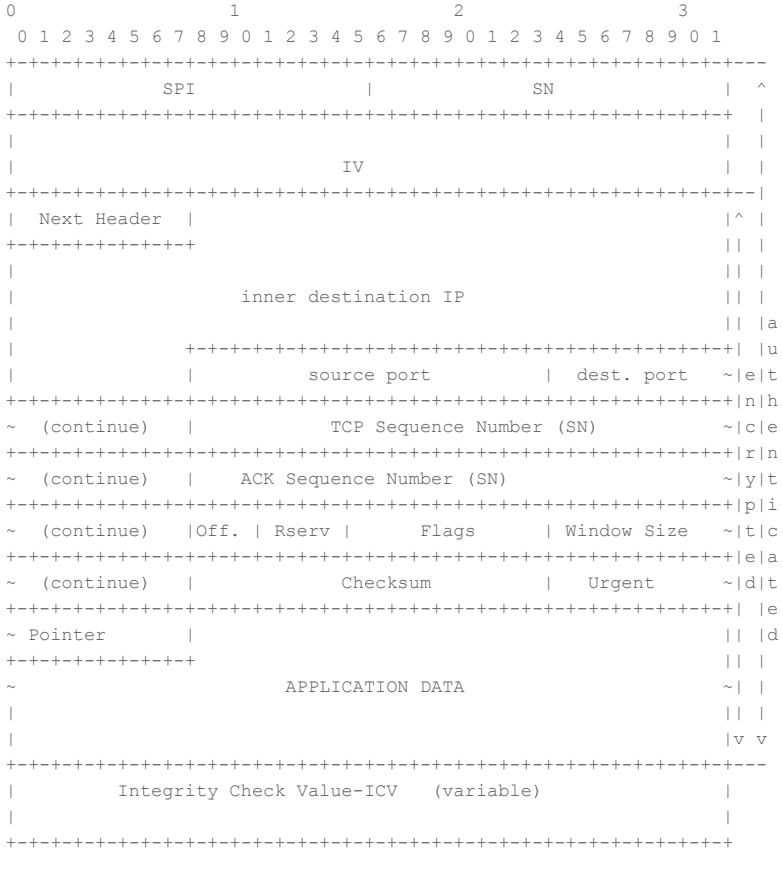


Example 2: VPN

EHC Rule	Context Attribute	Value
ESP_SPI	esp_spi_lsb	2
ESP_SN	esp_sn_lsb	2
	esp_sn_gen	
ESP_NH		
ESP_PAD	esp_align	8
IP6_OUTER	ip6_tcfl_comp	
IP6_LENGTH		
IP6_HL_OUTER	ip6_hl_comp	
IP6_SRC	ip6_src	in SA

Example 2: VPN (32 bytes)





Discussion Compression MAY NOT always apply

Problem:

- Compression MAY NOT always apply (IP fragmentation, UDP options….)
- We need to signal inband “UNCOMPRESSED” or “COMPRESSED”

Solutions:

- ADD an additional new bit (byte)
- Reuse existing field: (Protocol, Next Header) or SPI

We chose SPI:

- 2 SA one for ESP compressed with EHC and the other for ESP uncompress.
- New IPsec mode: EHC_COMPRESSED
    - Should it be negotiated separately versus implicitly derived from an EHC Strategy

Current status

Drafts:

- draft-mglt-ipsecme-diet-esp-06 describes EHC / Diet-ESP
- draft-mglt-ipsecme-ikev2-diet-esp-extension-01 describes the negotiation via IKEv2

We believe the documents are ready for adoption!

Publication/implementation (Contiki)

“Diet-ESP: IP layer security for IoT”

Thanks you!