Compact Format of IKEv2 Payloads

draft-smyslov-ipsecme-ikev2-compact-00

Valery Smyslov svan@elvis.ru

IETF 97

Motivation

- Reducing size of IKEv2 messages would decrease power and network bandwidth consumption (important for IoT devices)
- Reducing size of IKE_SA_INIT messages would decrease chances of IP fragmentation

Existing Format Redundancy

- Many payloads contain substantial redundancy
 - Payload Length field occupies 2 bytes, while most payloads are shorter
 - most parameters occupy 2 bytes, while less than 256 values are defined
 - zero-filled RESERVED fields

Example: SA Payload on the right contains one Proposal with four Transforms:

- ENCR_AES_CBC (128 bits)
- PRF_HMAC_SHA2_256
- AUTH_HMAC_SHA2_256_128
- 2048-bit MODP Group

Payload length is **48** bytes, among which **24** bytes are zeroes.

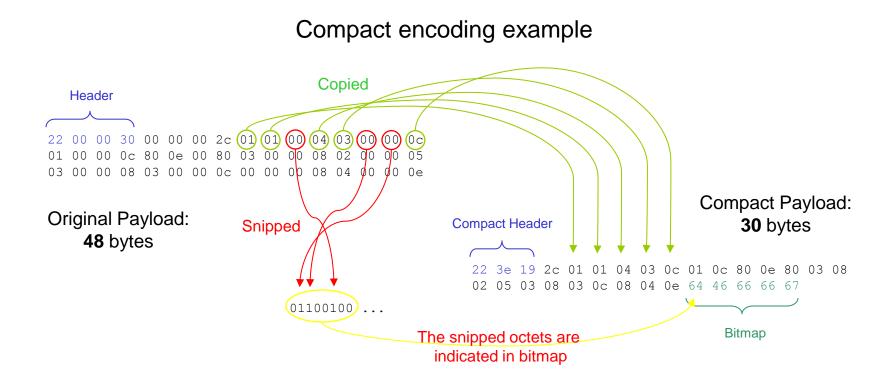
<u>Eile E</u>	dit ⊻iew	<u>Go</u> <u>C</u> ap	oture <u>A</u> nah	yze <u>S</u> tat	istics T	elephony	Tools Ir	ternals H	elp							
		🕷 🖻		281	°, 4	: 😜 🖗	₮ ⊉		0, Q	0, 🖻	2	M 😼	*	Ø		
Filte	er:							Sec. Expre	ssion	Clear		Apply		Save		
).	Time 1 10:		875669	Source	11.10	.199		stination).111.1	0.191		Prote ISA	ocol Le AKMP	ength 53	Info 8 IKE_	_SA_II	NIT
Tot	annat	Focuni	ity Asso	ociati	00 70	d Kov	Manage	mont D	rotoc	-01	10					
2.16		ayload:	0.001			a ron S										
E: ⊪ F M	lags: essage ength: ype Pa	n: 2.0 ge type 0x08 2 ID: 0 496 ayload:	: IKE_S ×000000 Securi	SA_INI 000 Ity As	T (34 socia) tion (,									
E F M L T	xchang lags: essage ength: ype Pa Next (0 Paylo Type (n: 2.0 ge type 0x08 2 ID: 0 496 ayload: payload = ad leng Payload	: IKE_S ×000000	GA_INI 000 Ity As Exchar al Bit osal 0	T (34 socia nge (: :: No1 (2) #) tion (34) t Crit [.] 1)	33)				_					
	xchang lags: essage ength: ype Pa Next p 0 Paylo Type Pa ype Pa	1: 2.0 ge type 0x08 2 ID: 0 496 ayload: payload Payload ayload:	: IKE_S ×000000 Securi d: Key I Critic gth: 48 d: Prop Key Ex	GA_INI 000 Exchar al Bit osal ((chang	T (34 socia nge (: :: Not (2) # e (34) tion (34) t Crit 1)	33) ical	08 00	00			1.2				>
E: F M L T 030 040	xchang lags: essage ength: ype Pa Next I 0 Paylo Type I ype Pa e1 C. 00 00	1: 2.0 ge type 0x08 2 ID: 0 496 2 ID: 0 4 2 ID: 0 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	: IKE_S ×000000 Securi d: Key I Critic. gth: 48 d: Prop. Key Ex Key Ex	GA_INI 000 Exchar al Bil osal (cchang 00 00 22 00	T (34 socia nge (: :: Not (2) # e (34) tion (34) t Crit 1) 1 00 21 30 00	33) ical 20 22 00 00	08 00 2c 01	01			·!				>
E: F Mi Li T T 030 040 050	xchang lags: essage ength: ype Pa 0 Paylo. Type I ype Pa eI C. 00 00 00 00	1: 2.0 ge type 0x08 2 ID: 0 2 496 ayload: payload payload ayload 10 00 0 00 00 0 00 00 4 03 00	: IKE_S x0000000 Securi d: Key I Critic. gth: 48 d: Prop Key Ex 5 00 00 0 01 00 0 00 00	GA_INI DOO Exchar al Bit osal (cchang 00 00 22 00 01 00	T (34 socia nge (: :: No1 (2) # e (34) 00) 00	tion (34) t Crit: 1) 00 21 30 00 0C 80	33) ical 20 22 00 00 00 00	2c 01 80 03	01 00		•••	· · I	· · · · · · · · · · · · · · · · · · ·			>
E F M L T	xchang lags: essage ength: ype Pa Next J 0 Paylo Type Pa ei C. 00 00 00 00	1: 2.0 ge type 0x08 2 ID: 0 496 ayload: payload payload 0 00 00 0 00 00 4 03 00 8 02 00 8 04 00	: IKE_S ×000000 Securi d: Key I Critic. gth: 48 d: Prop. Key Ex Key Ex	5A_INI 500 1 ty AS Exchar al Bit osal 0 cchang 00 00 22 00 01 00 03 00 28 00	T (34 socia nge (: :: No1 (2) # e (34) 00) 00) 00) 00) 00	tion (34) t Crit 1) 00 21 30 00 0C 80 08 03	33) ical 20 22 00 00 0e 00 00 00 0e 00	2 c 01 80 03 0 c 00 00 39	01		Ľ.					>

Compact Format Requirements

- The compact format must be generic and must be applicable to any payload, including not yet defined payloads
 - however, some payloads may have special format if it is justified
- The compact format must be easily converted to regular format and visa versa
 - existing parsing/composing code can be reused
- The compact encoding must never increase payload size
- The encoding/decoding algorithms must be simple and consume low resources

Generic Compact Format

Outline: snip zero octets from the payload data and append a bitmap that indicates which octets were snipped



Special Compact Formats

Special compact format is defined for:

- SA Payload
 - SA Payload grows quickly as more and more new transforms are defined and offered by initiators
- Notify Payload with Status Type Notification and no data
 - Exchange of such payloads is a common way to negotiate support for various protocol extensions, so initial IKEv2 messages grow up as more and more extensions are defined

Both payloads contain a lot of redundancy and can be effectively compacted.

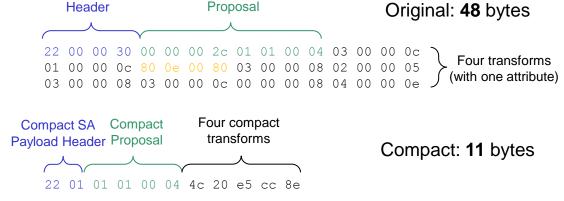
Compact SA Payload

Outline:

- Remove all RESERVED fields
- Remove Length fields in substructures (where they are unnecessary)
- Encode all currently defined transforms w/o attributes in one octet (both Transform Type and Transform ID)
- Encode currently defined Encryption transforms having Key Length attribute in two octets
- Leave possibility to encode arbitrary (even not yet defined) Transform Types
 and Transform IDs, as with regular format

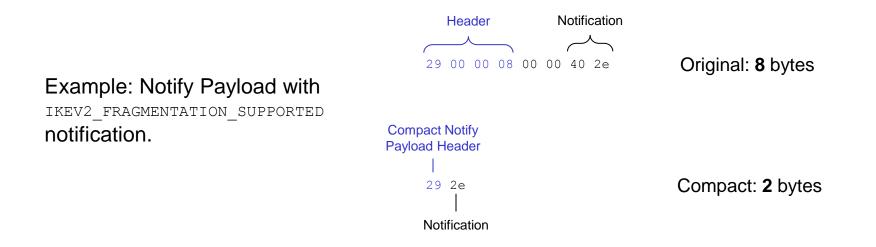
Example: SA Payload with one Proposal and four Transforms:

- ENCR_AES_CBC (128 bits)
- PRF_HMAC_SHA2_256
- AUTH HMAC SHA2 256 128
- 2048-bit MODP Group



Compact Notify Payload

Outline: encode notification in one octet (limited to first 256 status notifications) and omit all other fields from Notify Payload



Negotiation

Since compact format requires special parsing and is used in an initial IKEv2 exchange, it cannot be negotiated in a usual way – by exchange of Notify Payloads. Instead, a new exchange type ALT_IKE_SA_INIT is used in place of IKE SA INIT (with the same semantics).

 If Responder doesn't supports compact format, then she either replies with INVALID_SYNTAX notification or doesn't reply at all

Initiator may revert to IKE_SA_INIT exchange

• If Responder supports compact format, then she replies with ALT_IKE_SA_INIT response, which means that compact format is negotiated

Using

- Once negotiated the compact format can be used in any subsequent exchange
- Messages may contain both compact payloads and regular payloads
 - Generic Compact format is distinguished from regular payloads by non-zero bits in Payload Header's RESERVED field
 - Special Compact formats for SA Payload and Notify Payload have their own Payload Types
- Not all payloads are suitable for compact form
 - some payloads cannot be represented in compact form (e.g. if payload length exceeds 256 bytes)
 - some payloads don't benefit from compact form, because they usually contain random or pseudorandom data (e.g. Encrypted Payload or Nonce Payload)

Integration

Compact format can be easily integrated into existing IKE implementations

- Low complexity: ~200 lines of code in C++ for both encoder and decoder (including special format for SA and Notify payloads)
- Low resource consumption for encoding/decoding
- Can be implemented as pre/post process steps in message parsing/composing code

Thanks

- Comments? Questions?
- More details in the draft
- Please review and send feedback to author