1722 / 1722.1 ECC Encryption Considerations

Jeffrey Koftinoff
Apple Inc

2021/05/12

The Problem

- IEEE Std 1722.1 needs public key crypto (ECC) in order to transport session AES-SIV keys to the devices on the network.
- IEEE Std 1722-2016 Clause 17.3 defines how to use ECC Encryption using Elliptic Curve Integrated Encryption Scheme (ECIES) in “Ephemeral / Private” mode.
- This means that the sender encrypts a message in a manner where only the receiver can decrypt it, but the receiver can not authenticate the sender.
- See https://cryptobook.nakov.com/asymmetric-key-ciphers/ecies-public-key-encryption
- The AUTH commands in IEEE 1722.1 were designed around the assumption of “Private / Private” mode, where the receiver could authenticate the sender via the sender’s public key.
- It is possible to add new AUTH commands to IEEE 1722.1 which contain nonces that the Controller sends to the Responder and the Responder inserts into the reply to ensure that both ends of the conversation can be authenticated via the public keys of the other.

The Details

- IEEE Std 1722-2016 added Clause 17, “Encrypted Control Format”
- Clause 17.2.5 states:

> The **encrypted_payload** field contains the output of the ECC encryption algorithm variant as selected by the **enc** field, and is limited in length to 1488 octets.

- There is one variant, variant 1, defined in Clause 17.3:

> When the **enc** field value is ECC1, the **encrypted_payload** field contains the ECC1 plaintext payload (see 17.3.2) encrypted with elliptic curve cryptography as defined in subclause 11.3.2 of IEEE Std 1363a-2004 with the following scheme options:
>
> - The secret value derivation primitive shall be ECSVDP-DHC (subclause 7.2.2 of IEEE Std 1363- 2000), without compatibility with ECSVDP-DH
> - The method for encrypting the message shall be KDF2 (subclause 13.2 of IEEE Std 1363a-2004), using the hash function SHA-256 (subclause 14.1.3 of IEEE Std 1363a-2004)
> - The symmetric encryption scheme shall be AES-CBC-IV0 (subclause 14.3.2 of IEEE Std 1363a- 2004)
> - ECIES shall be used in DHAES (Diffie-Hellman augmented encryption scheme) mode (subclause D.5.3 of IEEE Std 1363a-2004)
> - The elliptic curve primitives shall be “X-coordinate only representation: EC2OSP-X and OS2ECP- X,” as defined in subclause 5.5.6.3 of IEEE Std 1363a-2004
>
> Decryption of the **encrypted_payload** field is achieved by using the algorithm defined in subclause 11.3.3 of IEEE Std 1363a-2004.

- The contents of encrypted_payload field is not formally detailed in IEEE Std 1722-2016.
- It is not clear from the above text that ECIES is used in Ephemeral/Private mode but it is; details are in IEEE 1363a Clause 11.3.1 - The “secret value” referenced above is the ephemeral private key.
- This is also why AES-CBC-IV0 is acceptable in this use case; the sender’s private key is never re-used.

Recommended fix for IEEE 1722.1

- Add AUTH_GET_NONCE command and response so that the Controller and Responder can share nonces. Each message is sent with Ephemeral/Private ECIES so they both get authenticated.
- Add AUTH_ADD_KEY_NONCE command which is like the AUTH_ADD_KEY command but also contains both Nonces collected during the AUTH_GET_NONCE command/response pair.

AUTH_GET_NONCE Command

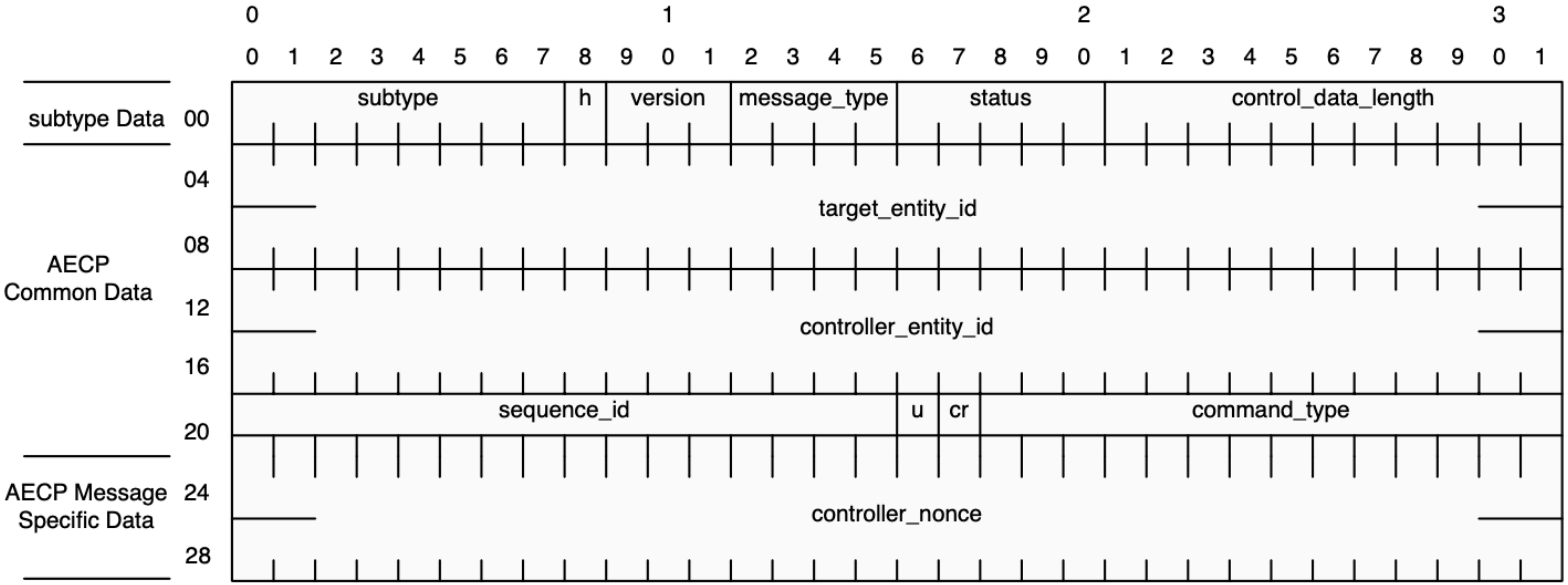
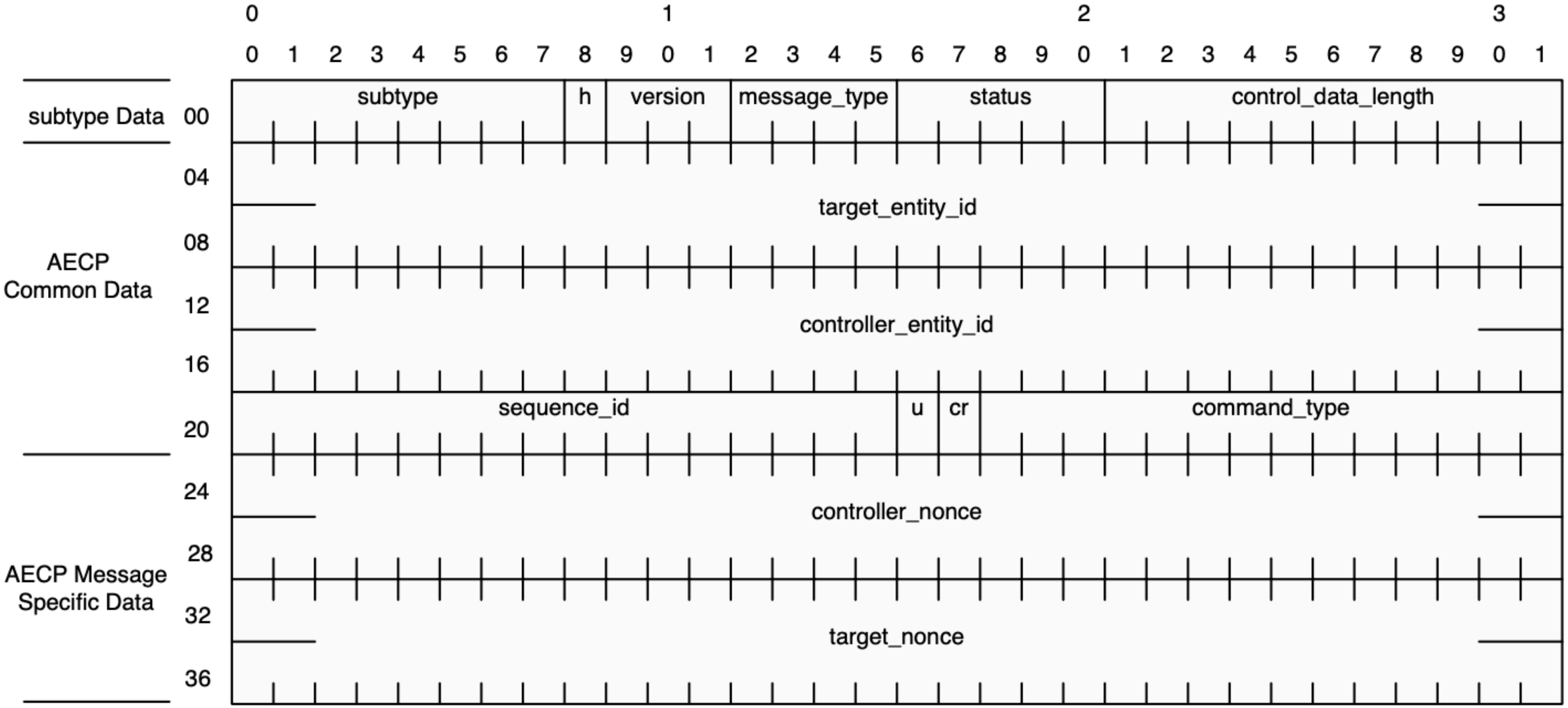


Figure 7-138—AUTH_GET_NONCE Command Format

AUTH_GET_NONCE Response



0 1 2 3

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

subtype Data 00: subtype | h | version | message_type | status | control_data_length

AECP Common Data

04

target_entity_id

08

12

controller_entity_id

16

20: sequence_id | u | cr | command_type

AECP Message Specific Data

24

controller_nonce

28

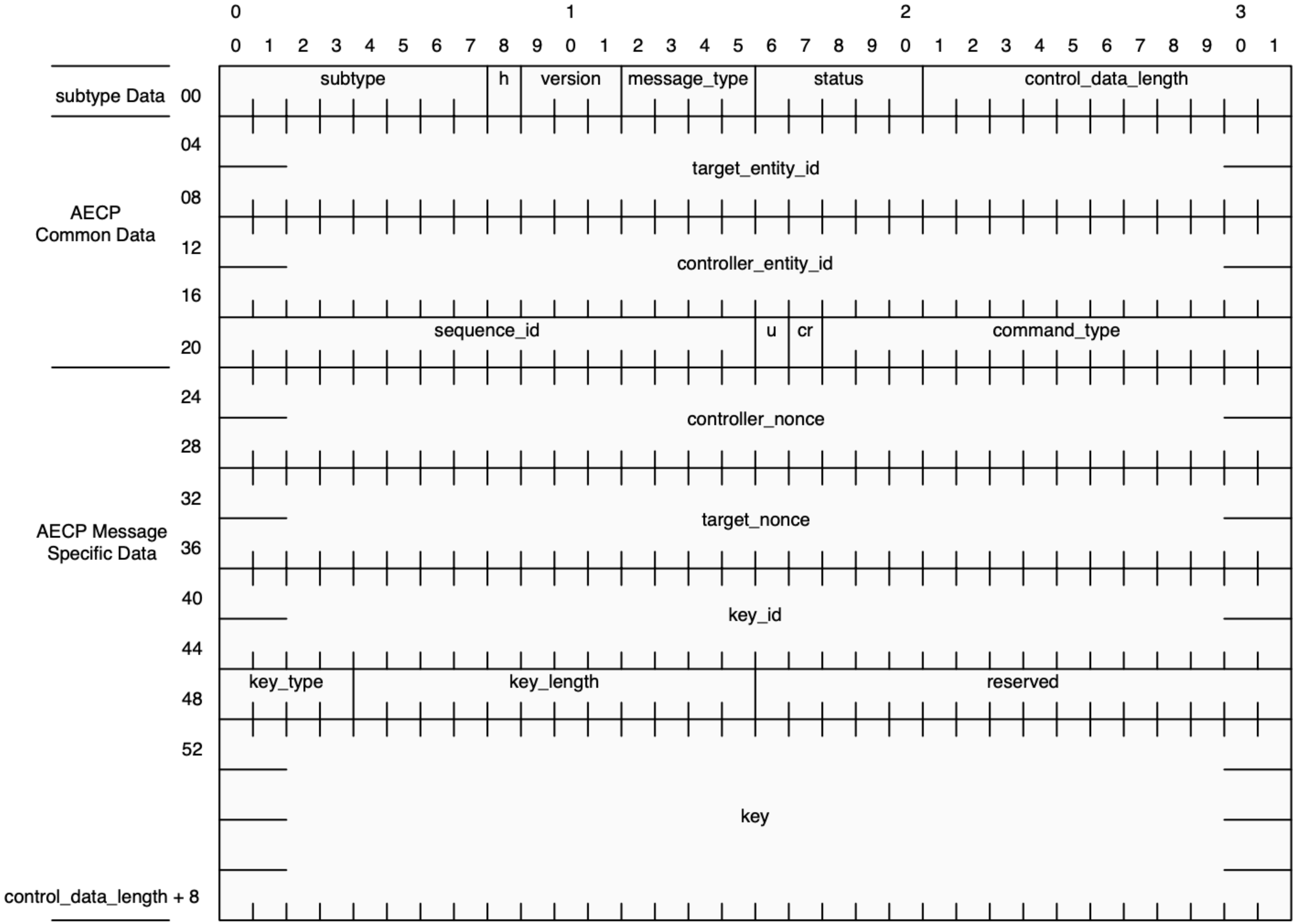
32

target_nonce

36

Figure 7-139—AUTH_GET_NONCE Response Format

AUTH_ADD_KEY_NONCE



subtype Data	00	subtype	h	version	message_type	status	control_data_length
AECP Common Data	04 08	target_entity_id					
	12 16	controller_entity_id					
	20	sequence_id	u	cr	command_type		
AECP Message Specific Data	24 28	controller_nonce					
	32 36	target_nonce					
	40 44	key_id					
	48	key_type	key_length	reserved			
	52 control_data_length + 8	key					

Figure 7-140—AUTH_ADD_KEY_NONCE Command Format