TLV Encryption and Packet Encapsulation

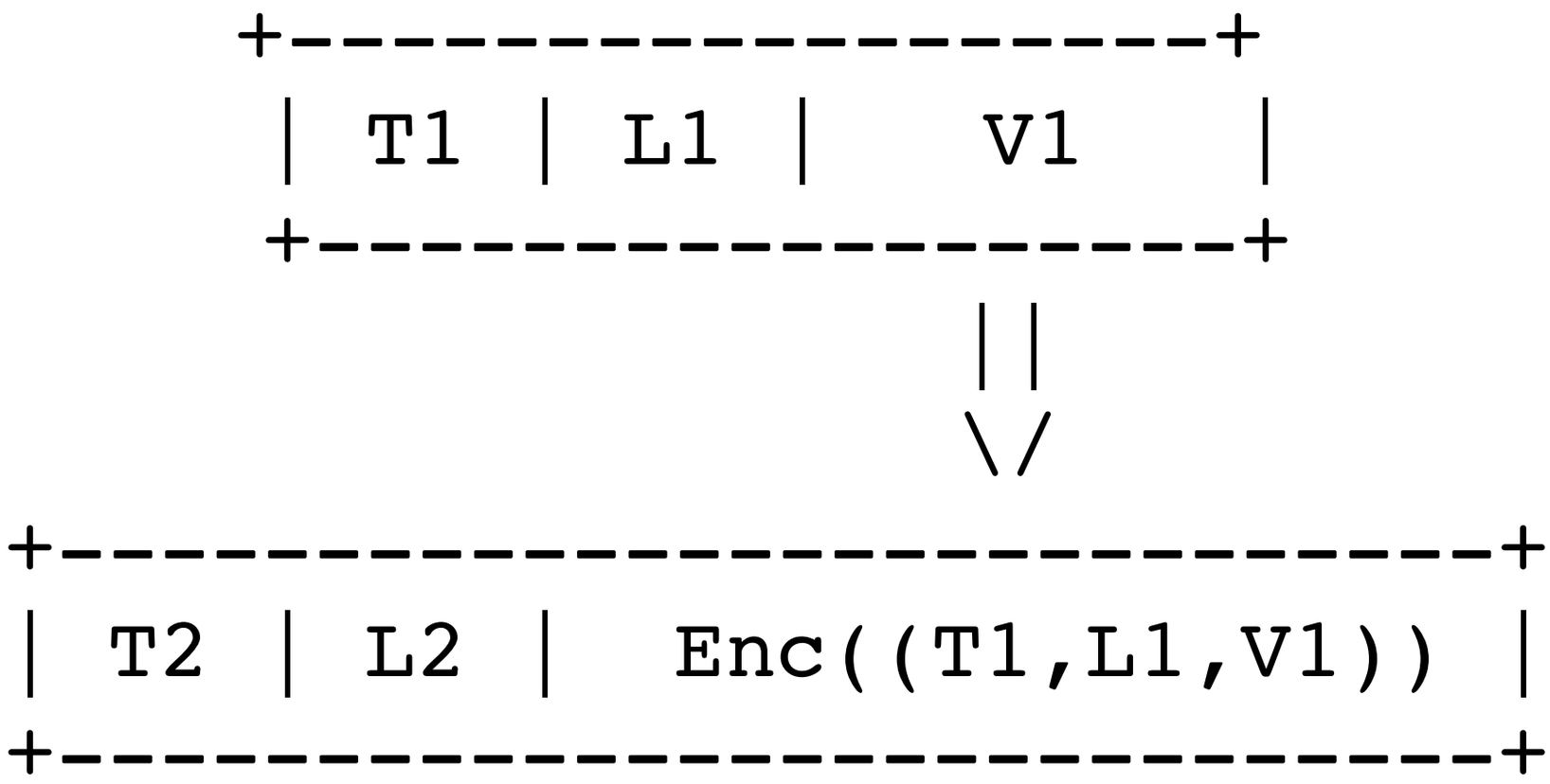
Overview

1. Specify an opaque TLV type to hold encrypted data

2. Specify use to encapsulate entire CCNx packets (interest and content object)

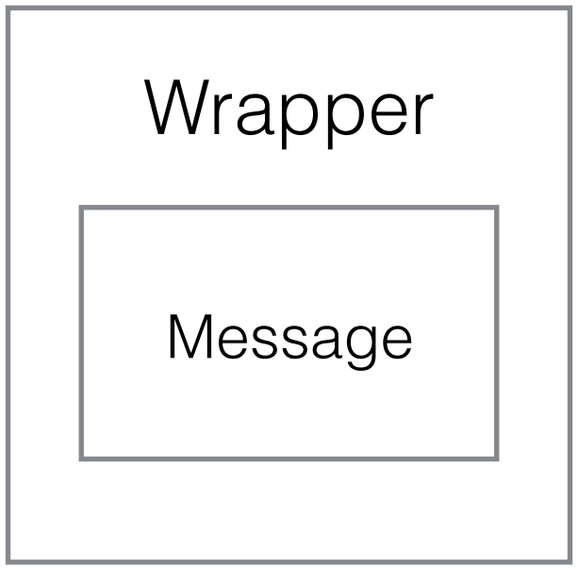
TLV Encryption

A new **`T_ENCAP`** TLV lets us do this:



The validation information (e.g., AES-GCM tag) is contained in a separate Validation TLV.

Packet Encapsulation



Packet Encapsulation

- Wrap interest and content objects in **`T_ENCAP`** TLVs
- Interests contain (in the name):
    - A routable prefix (**`/prefix/`**)
    - An identifier for the encapsulation decryption key and salt.
    - The encryption nonce (IV)
- Content objects contain:
    - An encapsulation name
    - A key identifier, salt, and nonce outside of the name (it may be separate from the content object)

Interest Encapsulation

```
Input: A plaintext CCNx Message TLV for an Interest I, and tuple
       (prefix, K, Salt, Nonce).
Output: An Interest I' with the encrypted I inside.

1. Create the Encapsulation Name EN as: /prefix/K/salt/Nonce.
2. Create a new Interest I' with the name EN, followed immediately by
   the TLV I contained inside a T_ENCAP TLV.
3. Create and append to I' a ValidationAlgorithm TLV with the
   T_VALIDATION_ALG type that specifies Interest encapsulation
   (**VALUE TBD**).
4. Encrypt all of I' using AES-GCM. The plaintext for this encryption
   procedure is only the V of the T_ENCAP TLV; The rest of message is
   the AAD. Let (C, T) be the output of this encryption process. Replace
   the V of the T_ENCAP TLV with C.
5. Create and append to I' a ValidationPayload that contains T.
6. Return I'.
```

Content Object Encapsulation

```
Input: An Interest I with name N, A plaintext CCNx Message TLV for a
       Content Object CO, and decryption information tuple
       (K, Salt, Nonce).
Output: A Content Object CO' with the encrypted CO inside.
```

1. Create the Encapsulation Name EN so that it matches N (the Interest Name).
2. Create a new Content Object CO' with the name EN, followed immediately by the TLV CO contained inside a T_ENCAP TLV.
3. Create and append to CO' a ValidationAlgorithm TLV with the T_VALIDATION_ALG type that specifies Content Object encapsulation (**VALUE TBD**), and a T_KEY_ID value that contains (K, Nonce, Salt).
4. Encrypt all of CO' using AES-GCM. The plaintext for this encryption procedure is only the V of the T_ENCAP TLV; The rest of message is the AAD. Let (C, T) be the output of this encryption process. Replace the V of the T_ENCAP TLV with C.
5. Create and append to CO' a ValidationPayload that contains T.
6. Return CO'.