CCNx Specification Updates

Marc Mosko
Palo Alto Research Center
ICNRG Interim Meeting Jan 14-15, 2016 Paris

Overview of Changes

- Hash agility — new formats for some fields
- New InterestReturn codes
- LCI name clarifications
- Optional header for carrying ContentObjectHash inside a trusted domain
- Nameless object inclusion

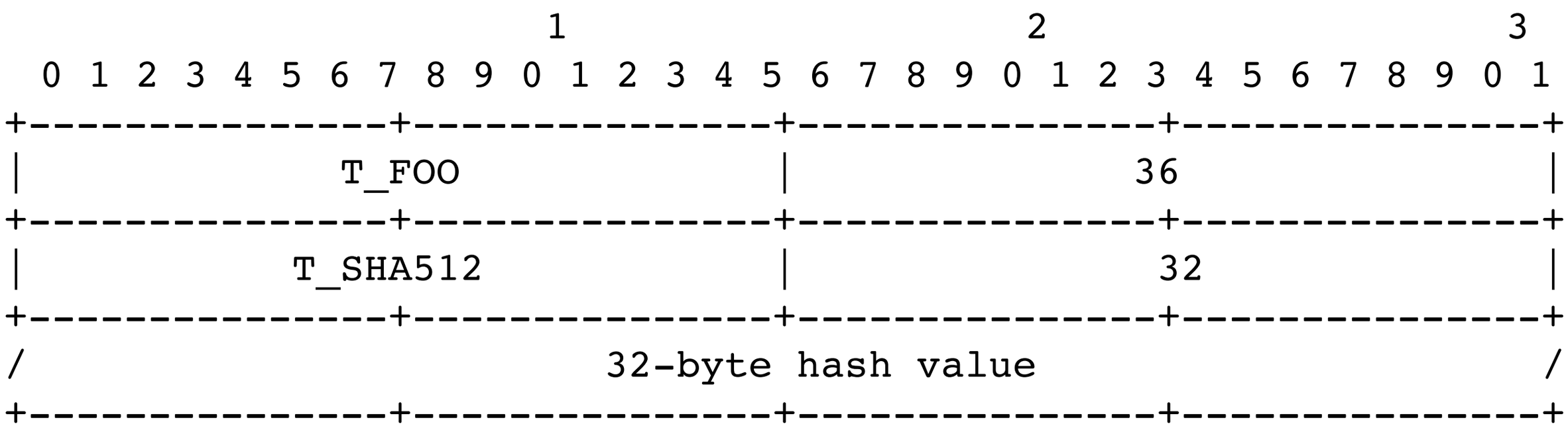
Hash Agility

- Purpose
  - Any field that uses a hash needs to be able to switch to a new hash
- Design choices
  - Use a new TLV Type (e.g. T_KEYID_SHA256 vs T_KEYID_SHA512-256)
  - Embed the type in the value (e.g. T_KEYID {T_SHA512-256 {hash value}}).

Hash Agility (2)

- We chose the embedded method because
  - For some fields intermediate nodes do not need to understand a new hash type to be able to binary compare fields (e.g. KeyId is not calculated at each node, it is just compared).
  - It leaves only 4 fields that a forwarder needs to know about (T_NAME, T_KEYID, T_KEYIDRESTR, T_OBJHASHRESTR) and it only needs to support the hash specified inside T_OBJHASHRESTR.
- If the Length of the TLV is less than the corresponding hash function length, the Value bytes are interpreted as the left-most bytes of the hash digest

Example



A SHA512-256 left truncation inside a type T_FOO

Hash Function Length Restrictions

Type	Abbrev	Lengths
%x0001	T_SHA-256	32
%x0002	T_SHA-512	64, 32
%x1000 - %x1FFF	n/a	any

Hash-Based Fields

- KeyIdRestriction and KeyId
  - Only significant for an end system, intermediate systems only do binary compare.
- ContentObjectHashRestriction
  - Must be supported by intermediate system, otherwise Interest is dropped and send InterestReturn.
- InterestPayloadID
  - Only possibly significant to end system, if it wants to verify the Interest Payload hash.

New Return Codes

- Unsupported ContentObjectHashRestriction
    - The Interest was dropped because it requested a Content Object Hash Restriction using a hash algorithm that cannot be computed.
- Malformed Interest
    - The Interest was dropped because it did not correctly parse (does not imply all systems need to validate).

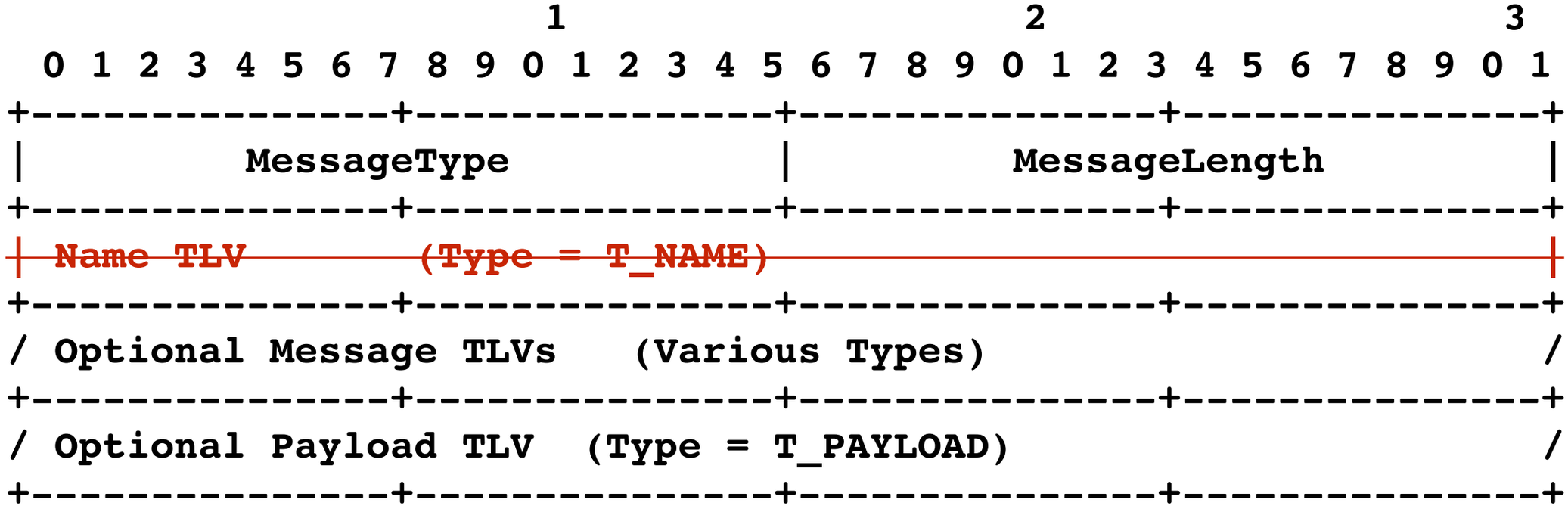
Name Clarifications

- T_NAME with length 4 and a T_NAMESEGMENT with 0 length corresponds to **`lci:/NAME=`**

- T_NAME with 0 length corresponds to **`lci:/`**

Nameless Objects

Nameless objects are content objects with **no name TLV**

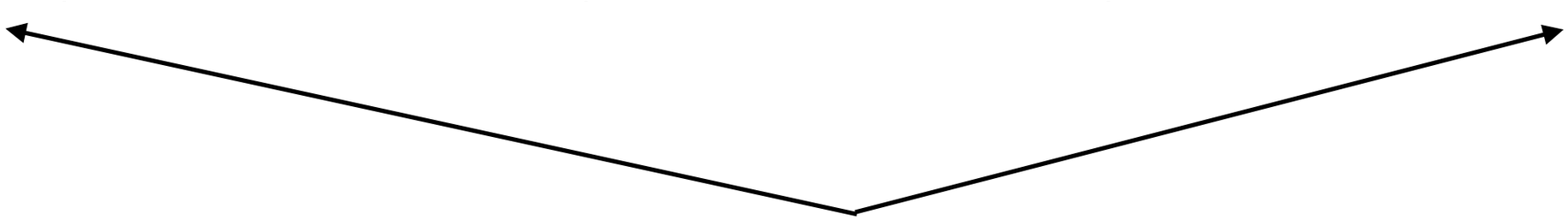


Content Object to Interest Matching

```
Interest
Ni = Name (always exists)
Ki = KeyIdRestr (may be empty)
Hi = COH Restr (may be empty)
```

```
ContentObject
No = Name (may be empty)
Ko = KeyId (may be empty)
Ho = Hash (always exists)
```

```
(∄ No | (Ni=No)) & (∄ Ki | (Ki=Ko)) & (∄ Hi | (Hi=Ho)) & (∃ No | ∃ Hi)
```



Terms specific to Nameless Objects