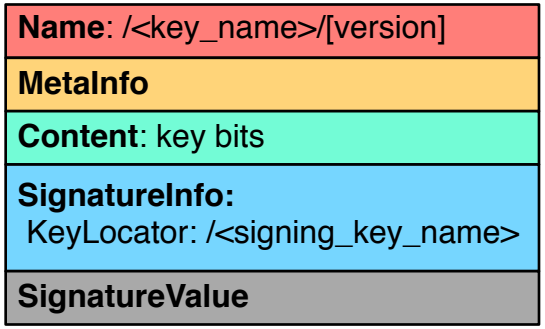
Multiple Signature

Certificate

- A data packet binds key name to key bits
  - signed by certificate issuer

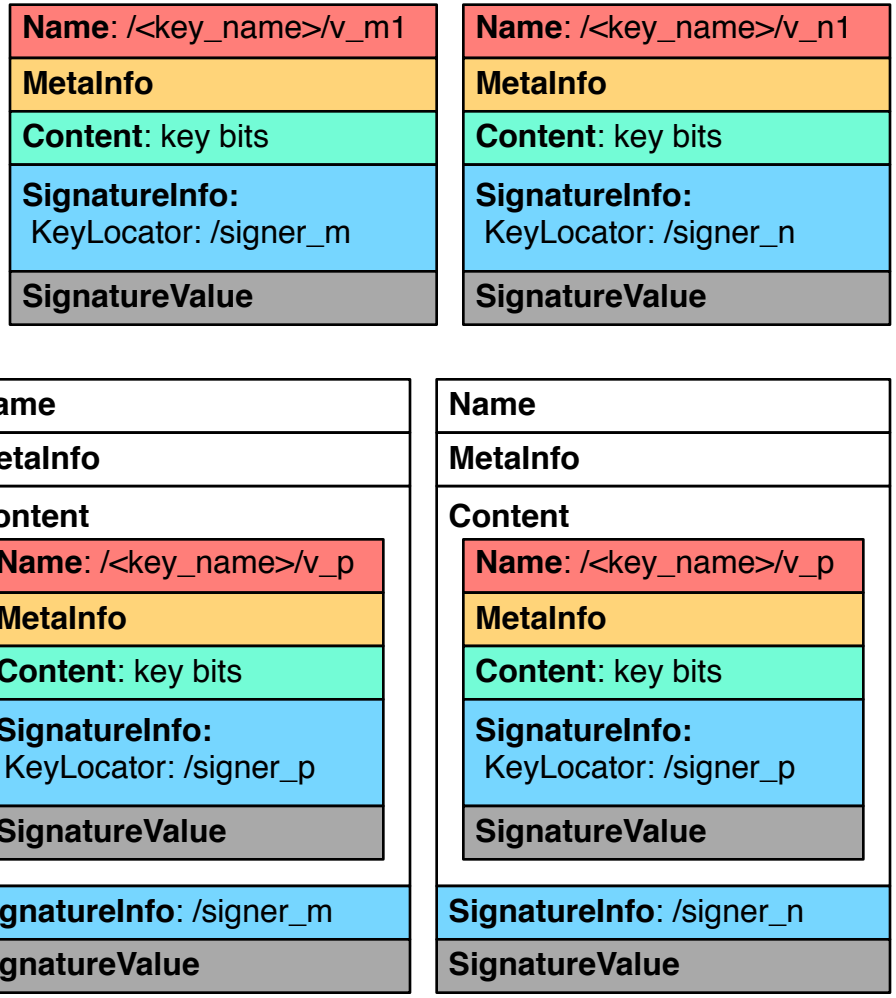


- Naming convention
  - /<key-name>/[version]
  - version
    - certificate issuer may replace its own key periodically
    - every time a new signing key is created, re-sign the binding between key name and key bits, leading to a certificate with new **version**

- Previously, we assume that there is only one issuer who can certify the binding between key name and key bits
  - version number is consistent from the issuer's perspective

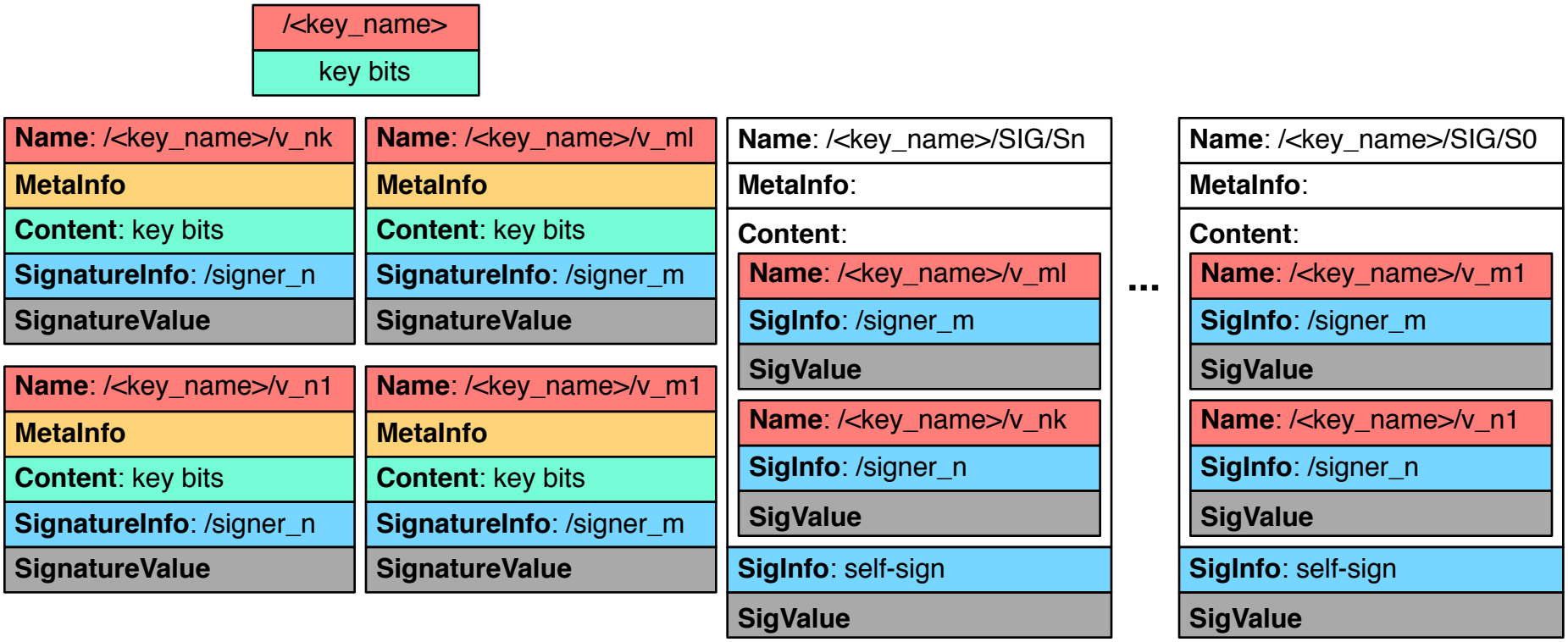
Multiple Signature

- Signature on the same (key name, key bits) binding?
  - how to maintain the version? or as long as version is consistent for each signer
  - v_m1 < v_m2 < v_m3 < …
  - v_n1 < v_n2 < v_n3 < …
- Signature on the same data packet?
  - encapsulation
  - who determine the inner version, sigInfo, and sigVal?
  - how to name the outer packet?
  - how to interpret such an encapsulation?



Design Options

- Option 3: signature bundles
  - group signatures into a separate data packet **signature bundle**
  - rely on naming convention to retrieve signature bundle
  - /<key_name>/SIG/[seqNo]



Signature Bundle

- Name:
  - /<key_name>/SIG/[SeqNo]
- Content:
  - a list of (Name, SignatureInfo, SignatureVal)
    - Name = key_name + signer specific version
  - (optional) full name of next signature bundle
- Signed by the key owner

- Retrieval
  - follow KeyLocator to retrieve key
    - KeyLocator does not include version number
  - (optionally) retrieve additional signatures