6LOWPAN

LoWPAN

- LoWPAN is Low-power Wireless Personal Area Networks (LoWPANs) composed of devices conforming to the IEEE 802.15.4.

  The devices are characterized by:

  - short range
  - low bit rate
  - low power
  - low cost

Characteristics of IEEE 80.15.4 networks(1)

- Small packet size(MTU:127 bytes)
- 16-bit short or IEEE 64-bit extended media access control (MAC) addresses
- Low data rates: 20 Kbps-250 Kbps
- Support of star and mesh topologies

Characteristics of IEEE 80.15.4 networks(2)

- Constrained devices
- Large number of deployed devices
- Ad-hoc networks
- Links are usually unreliable
- Nodes can be sleep state

6LoWPAN

- LoWPAN is a Low-power and Lossy Network (LLN) where the links interconnecting the nodes are IEEE 802.15.4 links.

- IPv6 protocol adaptation for LoWPAN

THE 6LoWPAN ADAPTATION LAYER

- The issues(1):
- IPV6 MTU(at network layer)-1280 bytes
- But LoWPAN MTU-127 bytes
- How to send ???
- **Need Packet fragmentation and reassembly**

- Why not increase MTU of LoWPAN?
- Handle limited buffering capacity and to limit packet error rate given high bit error rate

THE 6LoWPAN ADAPTATION LAYER(2)

- MTU-127 bytes
  - Maximum MAC frame overhead:25 bytes
  - MAC security header: 21 bytes
- **Only 81 bytes** (127 bytes – 25- 21 = 81) for the data payload (**IPv6 packets**).
- IPV6 header – 40 bytes
- TCP header-20 bytes

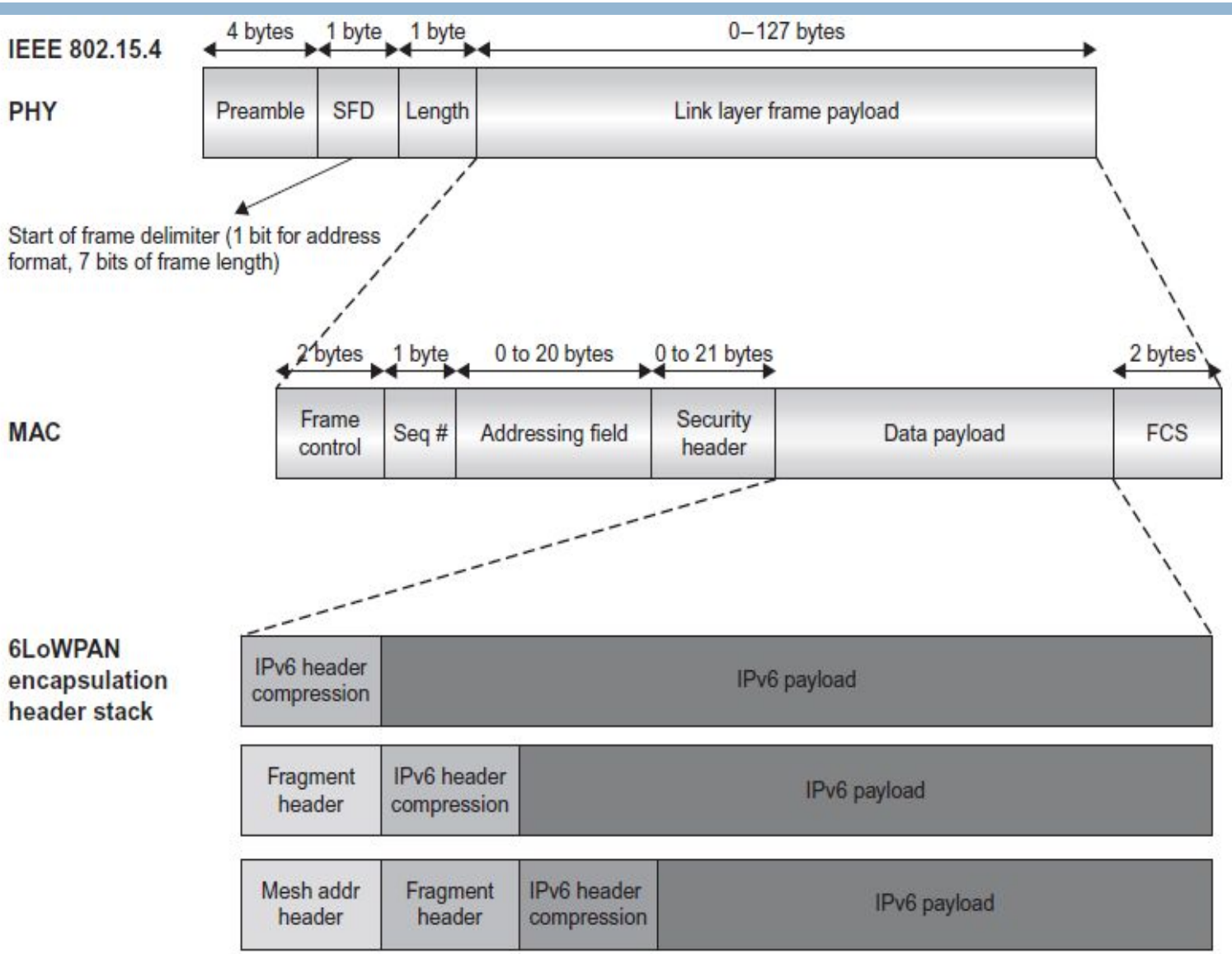
So, payload for application layer: 81-40-20=21 bytes !!!!

Need for header compression

Services of 6LoWPAN adaptation layer

- Packet fragmentation and reassembly
- Header compression
- Link layer (layer 2) forwarding when multi-hop is used by the link layer

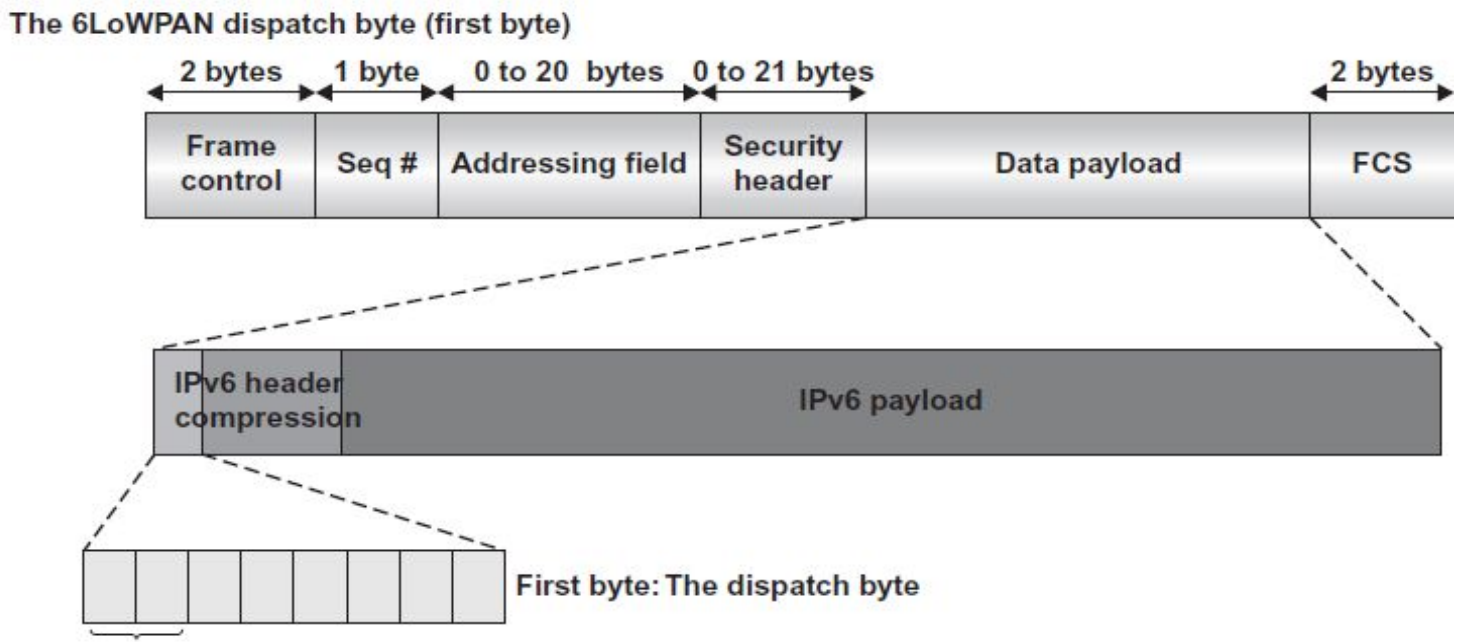
6LoWPAN encapsulation header stack



6LoWPAN encapsulation header stack

- Header stacking (headers are added only when needed).
- “ encapsulation header stack ” precedes each IPv6 datagram.
- Three headers:
  - a mesh addressing header
  - fragment header
  - IPv6 header compression header

Dispatch byte



00	Not a 6LoWPAN frame
01	IPv6 addressing header
10	Mesh header
11	Fragmentation header (6 lower bits are 100xxx)

Dispatch byte

First byte: The dispatch byte

Pattern	Header type
00 xxxxxx	NALP - not a LoWPAN frame
01 000001	IPv6 - uncompressed IPv6 addresses
01 000010	LOWPAN_HC1-LOWPAN_HC1 compressed IPv6
01 000011	reserved - reserved for future use
...	reserved - reserved for future use
01 001111	reserved - reserved for future use
01 010000	LOWPAN_BCO - LOWPAN_BCO broadcast
01 010001	reserved - reserved for future use
...	reserved - reserved for future use
01 111110	reserved - reserved for future use
01 111111	ESC - additional dispatch byte follows
10 xxxxxx	MESH - Mesh header
11 000xxx	FRAG1 - fragmentation header (first)
11 001000	reserved - reserved for future use
...	reserved - reserved for future use
11 011111	reserved - reserved for future use
11 100xxx	FRAGN - fragmentation header (subsequent)
11 101000	reserved - reserved for future use
...	reserved - reserved for future use
11 111111	reserved - reserved for future use

Mesh Addressing Header

- Support multi-hop “ routing ” at the link layer using link layer addresses.

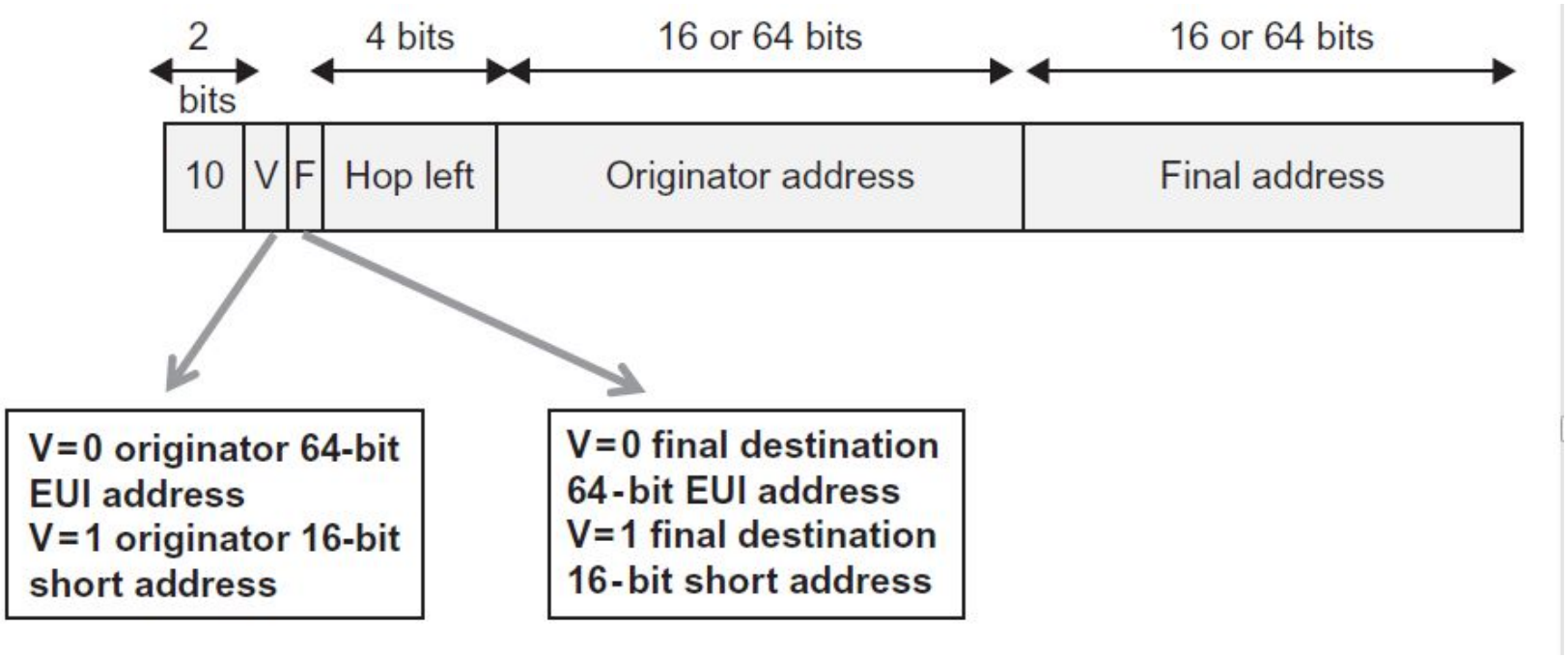
  Only FFDs can act as router, not RFDs.

  Source-Originator

  Destination-Final address

Dispatch byte should start with 10

Mesh addressing header



Hop left decremented at each intermediate node by 1, packet discarded if this field become 0.
Maximum value 15, addition deep hop field is added if more than 15 hops are equied

How packet forwarding works??

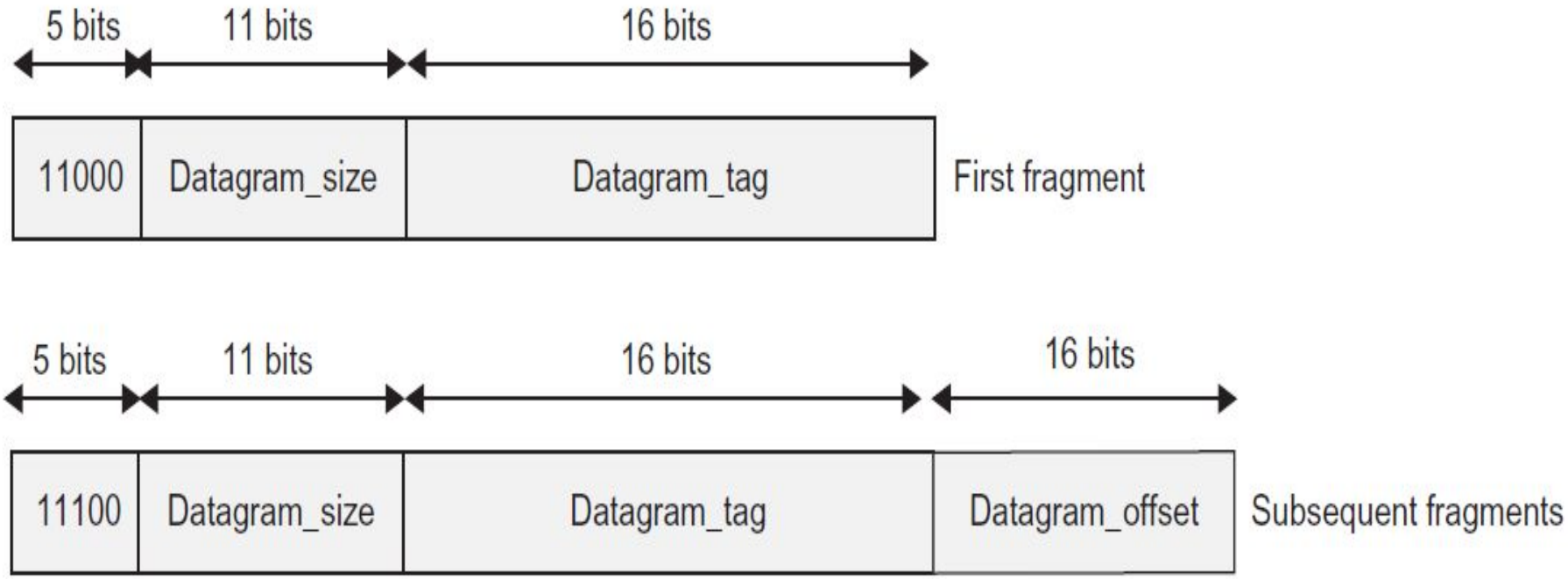
- Node A ---- Node B --- Node C
- Originator Final address

- **When a packet is transferred from A to B**
- Originator of header – Address of A
- Final address of header- Address of C
- Source address of the IEEE 802.15.4 frame - A
- Destination address of the IEEE 802.15.4 frame - B

How packet forwarding works??

- Node A ---- Node B --- Node C
- Originator Final address
- **When a packet is transferred from B to C**
- **Hop left is decremented by 1, If 0 discarded**
- Originator of header – Address of A
- Final address of header- Address of C
- Source address of the IEEE 802.15.4 frame - B
- Destination address of the IEEE 802.15.4 frame - C

Fragment header



Fragmentation

- Fragment sizes are expressed in units of 8 bytes

- **datagram_size**: To indicate the size in 8-byte units of the original IPv6 packet
- **datagram_tag:** uniquely identify the fragmented frame and must be identical for all link fragments.
- **datagram_offset:** indicates the offset in 8-byte units from the beginning of the payload datagram
- Use of **reassembly timer**