6LoWPAN

IPv6 over IEEE 802.15.4

- IEEE 802.15.4 is great for low-power wireless PANs
- IPv6 is great for the Internet of Things
  - Developers know IP (and higher protocols) well
  - All kinds of link technologies supported
  - Large address space
- Why IPv6 on 802.15.4?
  - We would get all the advantages of IPv6
  - IP-based border routers can connect with other IP networks (no complex gateways needed to translate protocols)

IPv6 over IEEE 802.15.4: Challenges

Can we run IPv6 over 802.15.4 networks?

- 802.15.4:
  - Maximum frame size 127 bytes
  - Maximum rate: 250kbit/s
- IPv6:
  - Minimum MTU: 1280 bytes
    → Datagram fragmentation needed
  - Headers:
    - 802.15.4 frame header up to 25 bytes (without security)
    - IPv6 40 bytes
    - UDP 8 bytes, TCP 20 bytes
  - Some IPv6 mechanisms (such as neighbor discovery) use link-local multicast/broadcast: How to do that in a mesh network?

Refresher: IPv6 Header

Fixed header format

<table>
<tr><th>Offsets</th><th>Octet</th><th colspan="8">0</th><th colspan="8">1</th><th colspan="8">2</th><th colspan="8">3</th></tr>
<tr><th>Octet</th><th>Bit</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>20</th><th>21</th><th>22</th><th>23</th><th>24</th><th>25</th><th>26</th><th>27</th><th>28</th><th>29</th><th>30</th><th>31</th></tr>
<tr><th>0</th><th>0</th><td colspan="4">Version</td><td colspan="8">Traffic Class</td><td colspan="20">Flow Label</td></tr>
<tr><th>4</th><th>32</th><td colspan="16">Payload Length</td><td colspan="8">Next Header</td><td colspan="8">Hop Limit</td></tr>
<tr><th>8</th><th>64</th><td colspan="32" rowspan="4">Source Address</td></tr>
<tr><th>12</th><th>96</th></tr>
<tr><th>16</th><th>128</th></tr>
<tr><th>20</th><th>160</th></tr>
<tr><th>24</th><th>192</th><td colspan="32" rowspan="4">Destination Address</td></tr>
<tr><th>28</th><th>224</th></tr>
<tr><th>32</th><th>256</th></tr>
<tr><th>36</th><th>288</th></tr>
</table>

Wikipedia

Refresher: IPv6 Addressing

- 128 bit
- Unicast addresses:
  - 64 bit for routing prefix and subnet id
  - 64 bit for interface identifier (IID; can be directly generated from the interface’s MAC address)
  - Global: 2000:… - 3FFF:…
  - Link-local: FE80:0:0:0:<IID>
- Multicast: FF..:…

Refresher: IPv6 Neighbor Discovery (ND)

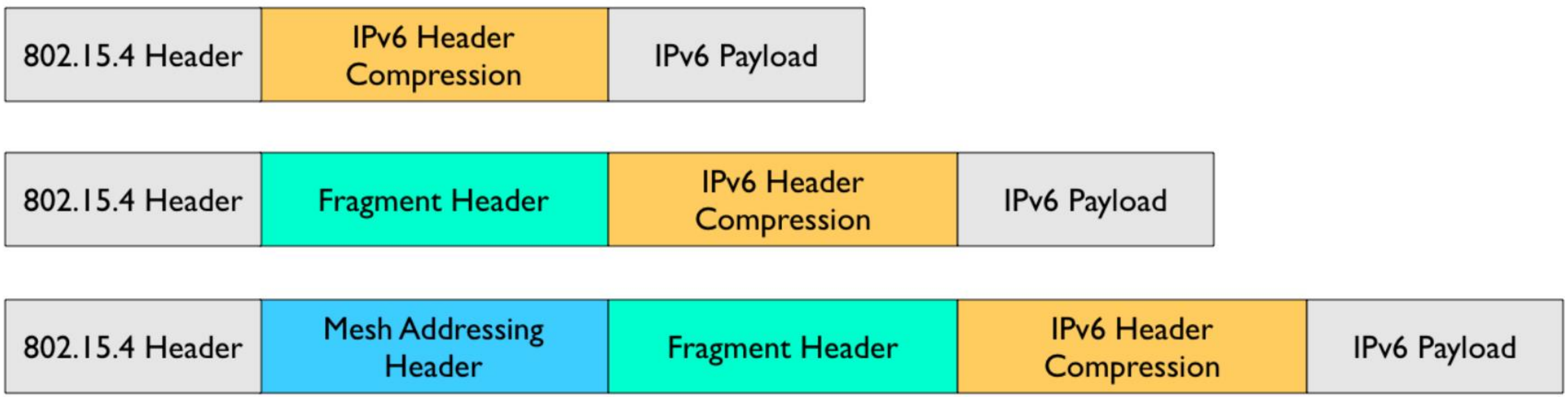
- Used for:
  - Router discovery (find router forwarding packets)
  - Prefix discovery (find addresses using the prefix)
  - Address resolution (replacing ARP for IPv4)
  - Duplicate address detection
  - Parameter discovery (MTU,…)
  - …
- Relies on multicast messages (sent to all link-local addresses)
  - In a mesh network, not guaranteed that all nodes can be reached

6LoWPAN

- An adaptation layer on 802.15.4 for IPv6 compliance
  - 802.15.4 network typically run in unslotted mode (CSMA/CA) with a mesh topology
- Provides:
  - Header compression: leave out unnecessary IPv6 header fields as much as possible
  - Fragmentation: IPv6 packets fragmented to frames
  - Link layer forwarding in mesh networks
- Not only for 802.15.4, also power-line communication etc.

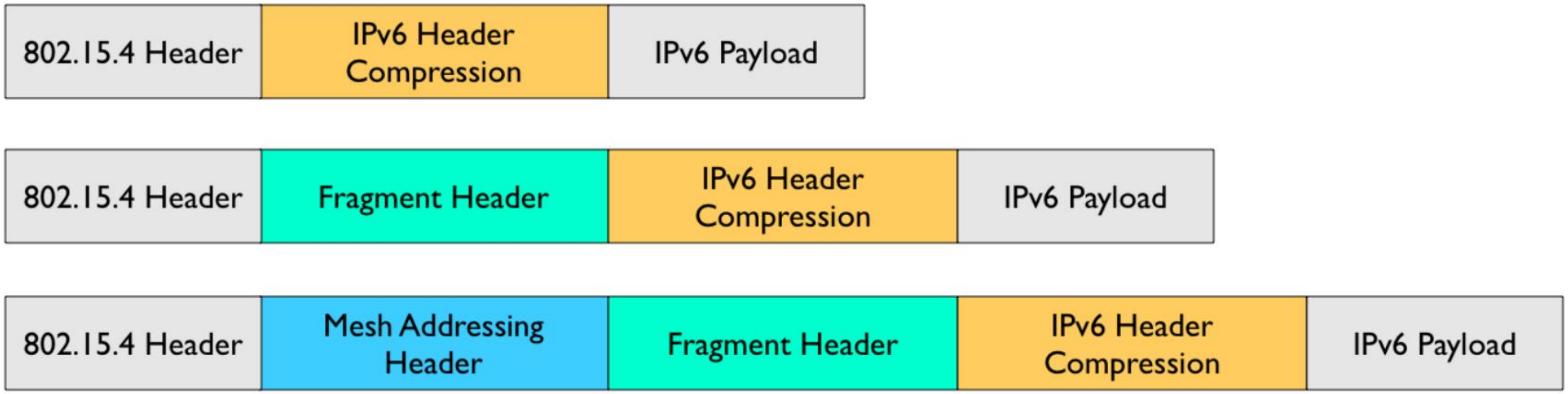
Stacked Headers

- Only include headers that are needed
- Some typical header configurations:



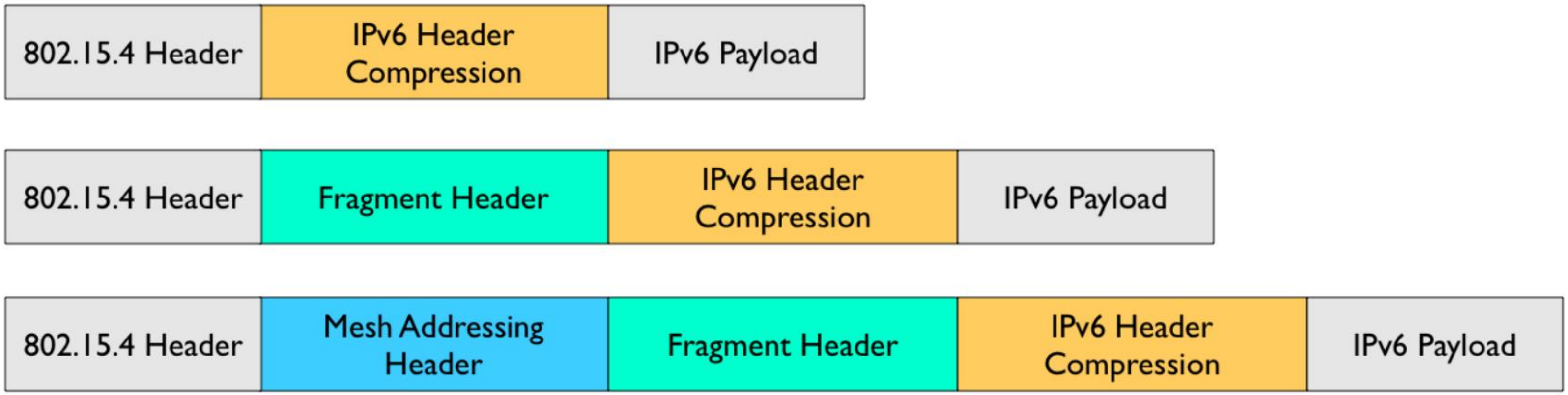
Hui et al. IPSO Alliance

Stacked Headers



- First byte (“dispatch byte”) defines header type
  - 00xxxxxx: not a 6LoWPAN frame
  - 01xxxxxx: IPv6 header (source/destination addr., etc.)
  - 10xxxxxx: Mesh header (for multi-hop forwarding)
  - 11xxxxxx: Fragment header (if packet is fragmented)

IPv6 Compression Header



- First byte of IPv6 Header:
  - 01 000001: uncompressed IPv6 header follows
  - 01 000010: HC1-compressed IPv6 header follows (RFC 4944)
  - 011 xxxxx: LOWPAN_IPHC (RFC 6282)

Addressing in HC1 Header

- Ideally:
  - First 64 bit of the IPv6 address are identical to the link-layer address of the 802.15.4 network
  - Last 64 bit (IID) of the IPv6 address are identical to the link-layer address (MAC address)

- In that case, all the information is already in the 802.15.4 frame header
  - No need to repeat them again in the HC1 header

HC1 Header

Basic Principle: assume default values unless the header explicitly says otherwise

1: IID derived from link layer address (resp. mesh addressing header)
0: IID provided uncompressed (64 bit)

00: uncompressed next-header field
01: UDP, 10: TCP, 11: ICMP

Hui et al. IPSO Alliance

1: link-local prefix assumed
0: prefix provided uncompressed (64 bit)

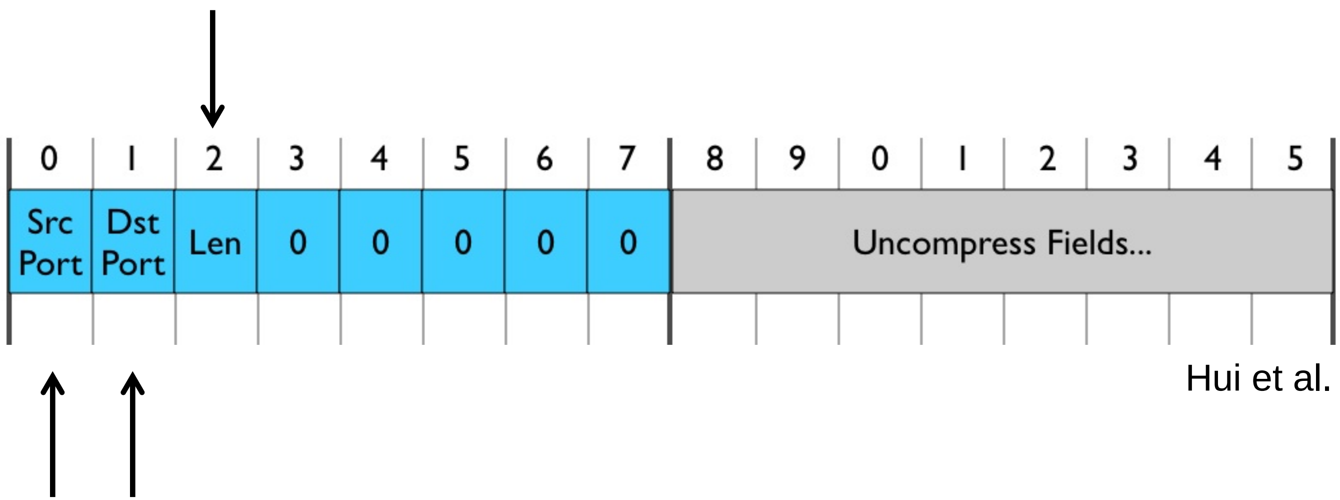
1: transport-layer header compressed
0: not compressed

0: Traffic Class (8 bit) and Flow Label (20 bit) are not compressed
1: Traffic Class and Flow Label are zero

UDP Header Compression with 6LoWPAN

- If HC2 bit is set, a HC2 byte immediately follows the HC1 byte (before the hop-limit field)
- +2 bytes for UDP checksum appended

0: length uncompressed
1: length derived from IPv6 payload length field



Hui et al. IPSO Alliance

0: port not compressed (16 bits)
1: port compressed (4 bits + 0xf0b0)

Drawbacks of HC1

- HC1+HC2 compression reduce UPD+IPv6 header to 7 bytes in the best case
  - Best case: link-local unicast addresses -> all information is already in the 802.15.4 frame
- Advantage is lost if the 6LoWPAN device has to communicate with devices external to the 6LoWPAN: Routable addresses needed
  - Full prefix has to be specified: 64 bit
  - Full IID has to be specified: 64 bit
  - Same for multi cast to IPv6 addresses

LOWPAN_IPHC compression (RFC 6282)

- Replaces HC1 compression
- Uses remaining 5 bits of the dispatch byte + 8 extra bits
  - Bits 3+4: for Traffic Class and Flow Label
    - 00, 01, 10: uncompressed or typical configurations
    - 11: Traffic Class and Flow Label set to zero
  - Bit 5:
    - 0: Next-Header uncompressed (8 bit)
    - 1: Next-Header field is compressed and LOWPAN_NHC compression is used for next header
  - Bit 6+7:
    - 00: Hop Limit is uncompressed
    - 01,10,11: Hop Limit is 1, 64 or 255

LOWPAN_NHC

- Compression of various possible next-header values
- Example: UDP compression (similar to HC2)
  - Checksum can be dropped (packets must be checked in application layer)
  - Port number compression:
    - Full 16 bit
    - Port number is 8 bit + 0xf000
    - Port number is 4 bit + 0xf0b0

LOWPAN_IPHC compression

- Remaining 8 bit of the IPHC header control address compression
  - Separately for source and destination address
- Address compression can be stateless (like in HC1) or stateful ("context-based")

LOWPAN_IPHC stateless compression

- Addresses can be defined in
  - Full 128 bit
  - Lower 64 bit
    - Upper 64 bit = network prefix from the link layer
  - Lower 16 bit
    - Upper 112 bit = network prefix from the link layer::0000:00ff:fe00
  - No bits (full compression like HC1), using the link-layer header fields

LOWPAN_IPHC stateless multicast

- Special compression of multicast destination addresses:
  - Full 128 bit
  - 48 bit address of the form FFxx::00xx:xxxx:xxxx
  - 32 bit address of the form FFxx::00xx:xxxx
  - 8 bit address of the form FF02::00xx

LOWPAN_IPHC stateful compression

- The Neighbor Discovery protocol in 6LoWPAN allows to share "context" information between nodes (RFC 6775)
- Context
  - An address prefix of length 0 to 128 bits
  - Can refer to remote addresses or local addresses
  - Has a specific lifetime

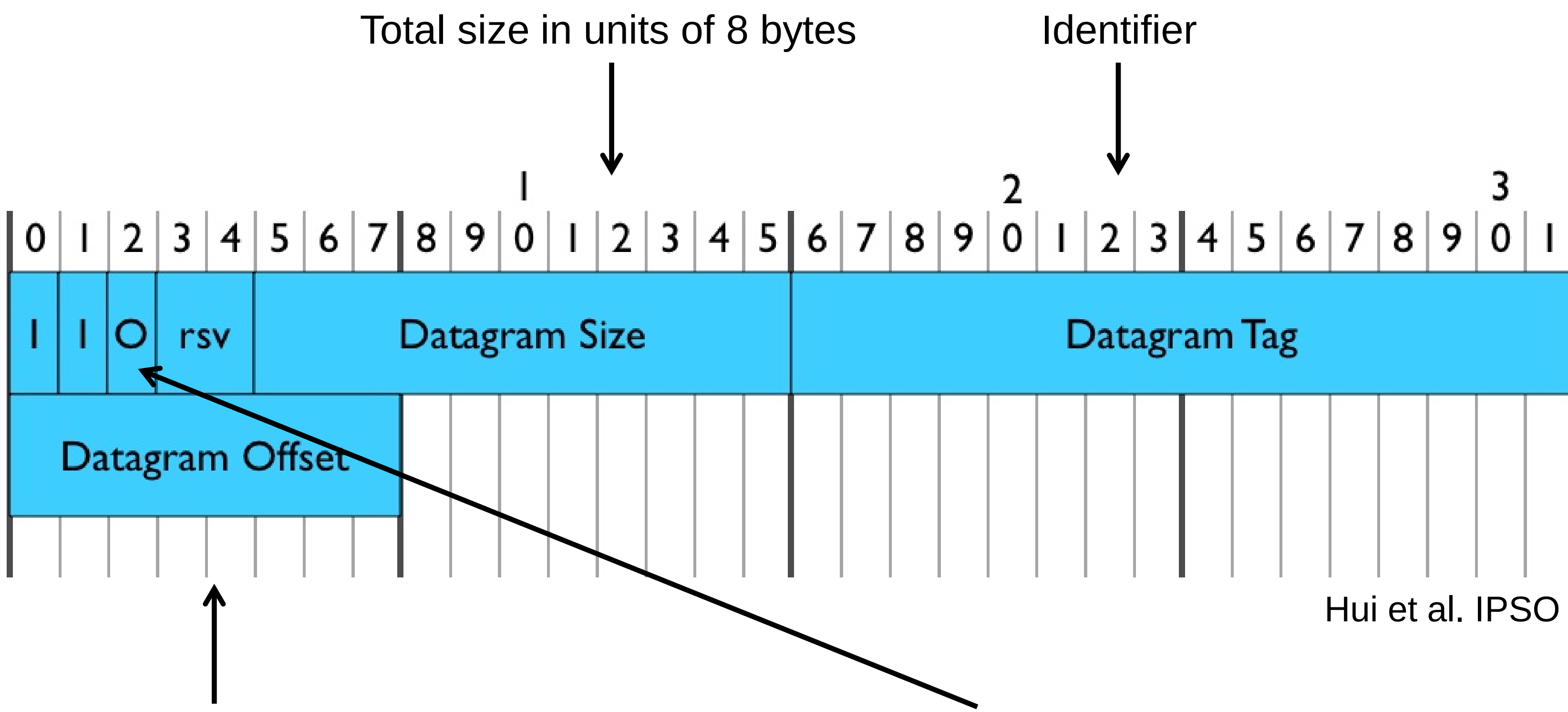
- Source and destination addresses in LOWPAN_IPHC can specify which context to use
  - No context (stateless)
  - Context 0
  - 1 out of 16 possible contexts (for source and destination, in an extra 8 bit field)

LOWPAN_IPHC stateful compression (2)

- Contexts allow to compress frequently used address prefixes
- Context compression is stateful: nodes have to store the context definitions

Fragment Header

- IPv6 minimum MTU is 1280
- 6LoWPAN fragments packets that do not fit into frame



Hui et al. IPSO Alliance

Offset in units of 8 bytes. First packet has no offset (bit 2 set to 0)

Mesh Addressing Header

- IEEE 802.15.4 frames can only be sent single hop
- If multi-hop mesh delivery is needed:
  - RFDs discover FDDs and send all their traffic to them
  - FDDs do forwarding and run a routing protocol to populate their routing tables

- With multi-hop, a packet has to carry two addresses:
  - The link address of the next hop
  - The link address of the original source and destination
    - Why?
    - Because the compressed IP header refers to it

Mesh Addressing Header

- 802.15.4 frame: specifies link-layer address of next hop
- Mesh Addressing Header specifies original/final link-layer address