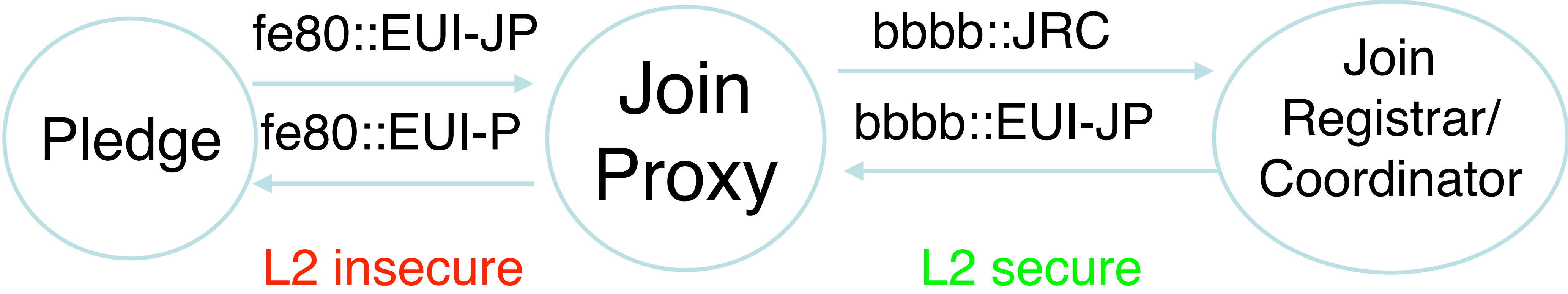
draft-ietf-6tisch-minimal-security

Mališa Vučinić, Inria
Jonathan Simon, Analog Devices
Kris Pister, UC Berkeley
Michael Richardson, Sandelman Software Works

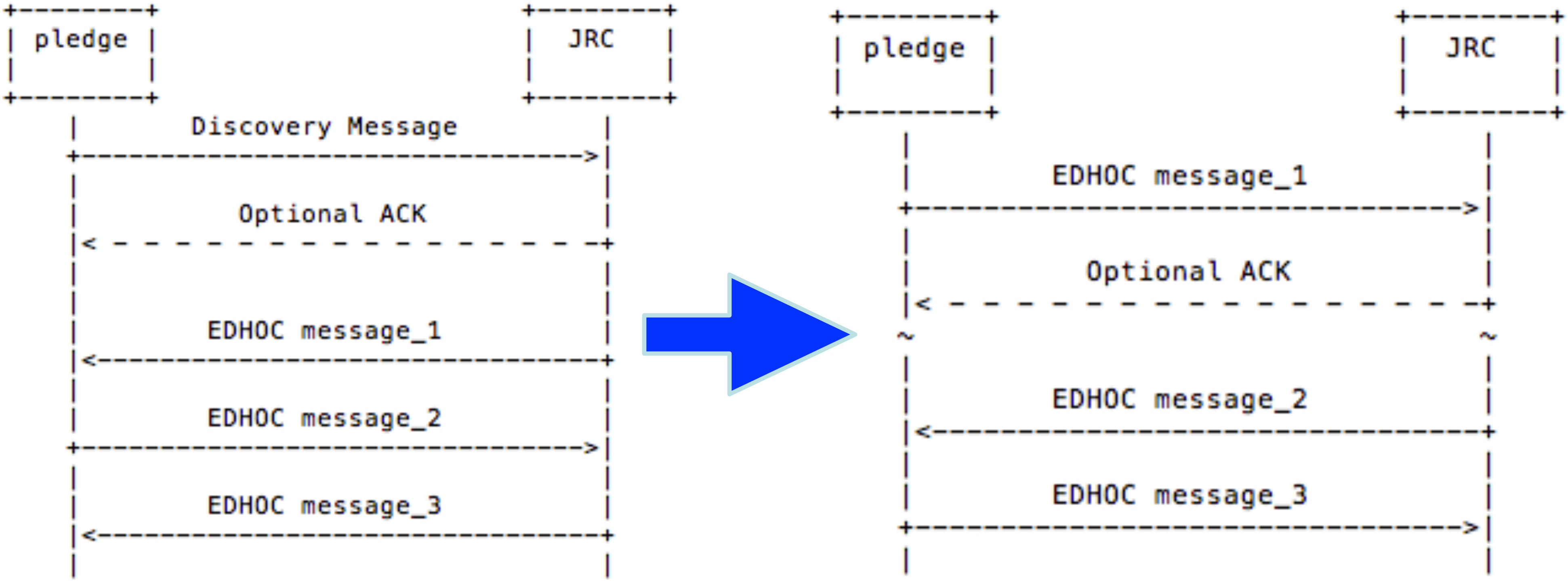
Status

- News
  - draft-ietf-6tisch-minimal-security-03
  - Published on June 15th 2017
- Implementation with PSKs in OpenWSN completed, Contiki ongoing
- Summary of updates in -03

Communication Overview



Update #1: Security Handshake



```
+----------+                     +----------+
| pledge   |                     |   JRC    |
|          |                     |          |
+----------+                     +----------+
     |        Discovery Message       |
     +------------------------------->|
     |                                |
     |          Optional ACK          |
     |<- - - - - - - - - - - - - - - -+
     |                                |
     |                                |
     |         EDHOC message_1        |
     |<-------------------------------+
     |                                |
     |         EDHOC message_2        |
     +------------------------------->|
     |                                |
     |         EDHOC message_3        |
     |<-------------------------------+
     |                                |
```

```
+----------+                         +----------+
| pledge   |                         |   JRC    |
|          |                         |          |
+----------+                         +----------+
     |                                    |
     |           EDHOC message_1          |
     +----------------------------------->|
     |                                    |
     |             Optional ACK           |
     |<- - - - - - - - - - - - - - - - - -+
     ~                                    ~
     |                                    |
     |           EDHOC message_2          |
     |<-----------------------------------+
     |                                    |
     |           EDHOC message_3          |
     +----------------------------------->|
     |                                    |
```

- Optional with PSKs
- Mandatory with asymmetric keys

Update #2: How pledge learns JRC address

- Join Proxy (JP) statelessly forwards to JRC
- How JP knows the address of JRC?
  - Learns at join time when it acted as a pledge
  - Join Response now contains the address
  - Omitted if JRC is co-located with DAG root, implied from DODAG ID
  - Assumption: DAG root pre-configured with the address

Update #3: Mandatory to Implement Algorithms

- AEAD algorithm:
  - AES_CCM_16_64_128 from COSE
  - 8 byte authentication tag
  - Corresponds to 802.15.4 CCM* in nonce length
- Hash:
  - SHA-256
- Asymmetric:
  - P-256 Elliptic Curve (secp256r1)
  - ECDSA with SHA-256 signature algorithm

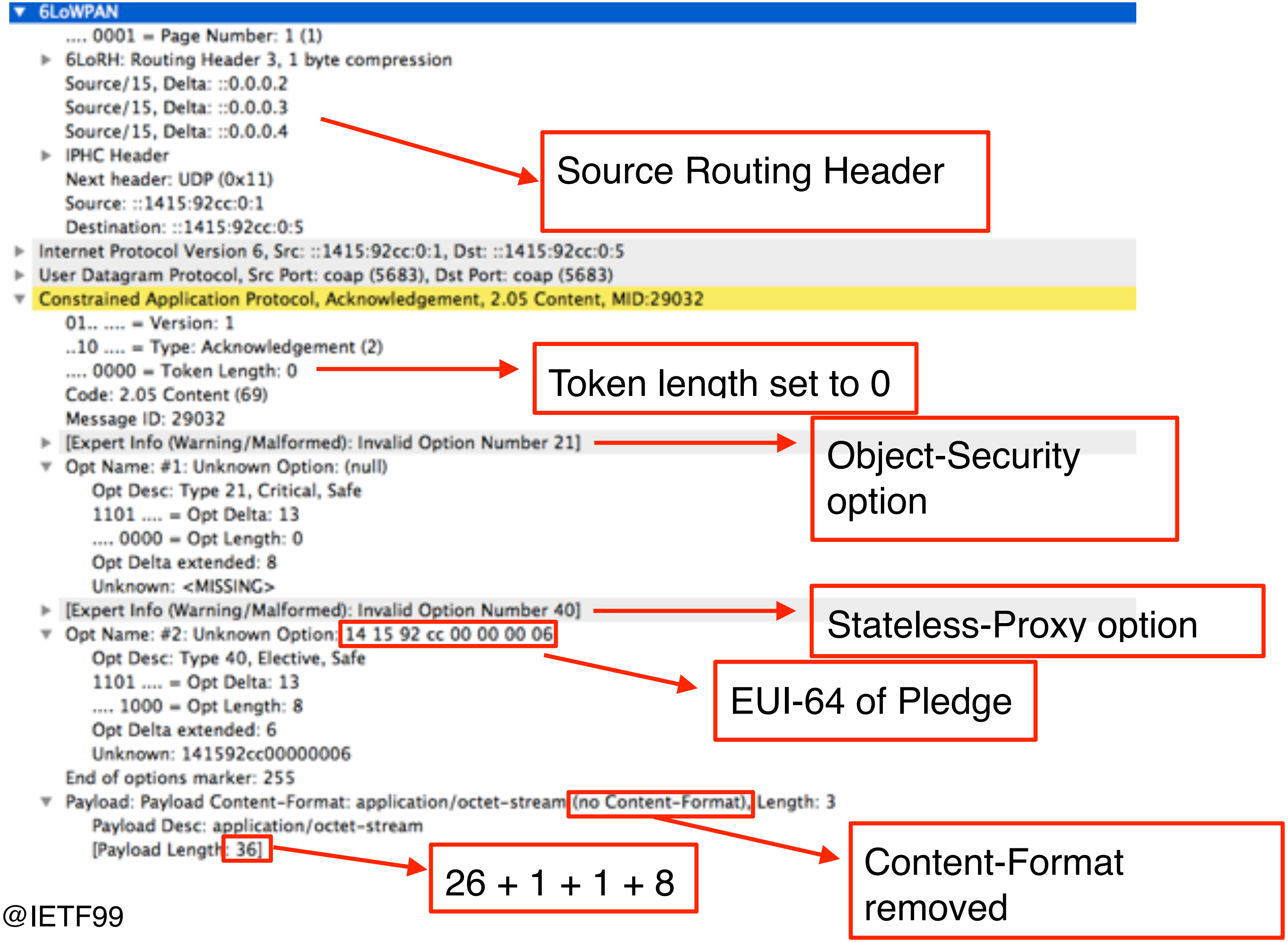
Implementation Status

- In OpenWSN ecosystem with Pre-Shared Keys:
  - draft-ietf-core-object-security-03 in Python
  - draft-ietf-6tisch-minimal-security-03 in Python (JRC)
  - draft-ietf-core-object-security-03 in C
  - draft-ietf-6tisch-minimal-security-03 in C (Pledge and Join Proxy)
- In Contiki:
  - draft-ietf-core-object-security-03
  - draft-ietf-6tisch-minimal-security-03 (ongoing)
  - draft-selander-ace-cose-ecdhe-07 (ongoing)

Implementation Experience

- Issue #1: Problems to fit Join Response in 127 bytes with multiple hops
  - Bottleneck is the link from DAG root to first hop
  - Due to the source routing header, there is a limit on max depth of the network without fragmentation
- Issue #2: Policy by which JP should accept insecure L2 frames from pledges
- Additional clarifications in the document needed on hooks to lower layers

Issue #1: Packet size



```
▼ 6LoWPAN
      .... 0001 = Page Number: 1 (1)
    ▶ 6LoRH: Routing Header 3, 1 byte compression
      Source/15, Delta: ::0.0.0.2
      Source/15, Delta: ::0.0.0.3
      Source/15, Delta: ::0.0.0.4
    ▶ IPHC Header
      Next header: UDP (0x11)
      Source: ::1415:92cc:0:1
      Destination: ::1415:92cc:0:5
▶ Internet Protocol Version 6, Src: ::1415:92cc:0:1, Dst: ::1415:92cc:0:5
▶ User Datagram Protocol, Src Port: coap (5683), Dst Port: coap (5683)
▼ Constrained Application Protocol, Acknowledgement, 2.05 Content, MID:29032
      01.. .... = Version: 1
      ..10 .... = Type: Acknowledgement (2)
      .... 0000 = Token Length: 0
      Code: 2.05 Content (69)
      Message ID: 29032
    ▶ [Expert Info (Warning/Malformed): Invalid Option Number 21]
    ▼ Opt Name: #1: Unknown Option: (null)
        Opt Desc: Type 21, Critical, Safe
        1101 .... = Opt Delta: 13
        .... 0000 = Opt Length: 0
        Opt Delta extended: 8
        Unknown: <MISSING>
    ▶ [Expert Info (Warning/Malformed): Invalid Option Number 40]
    ▼ Opt Name: #2: Unknown Option: 14 15 92 cc 00 00 00 06
        Opt Desc: Type 40, Elective, Safe
        1101 .... = Opt Delta: 13
        .... 1000 = Opt Length: 8
        Opt Delta extended: 6
        Unknown: 141592cc00000006
      End of options marker: 255
    ▼ Payload: Payload Content-Format: application/octet-stream (no Content-Format), Length: 3
        Payload Desc: application/octet-stream
        [Payload Length: 36]
```

Source Routing Header

Token length set to 0

Object-Security option

Stateless-Proxy option

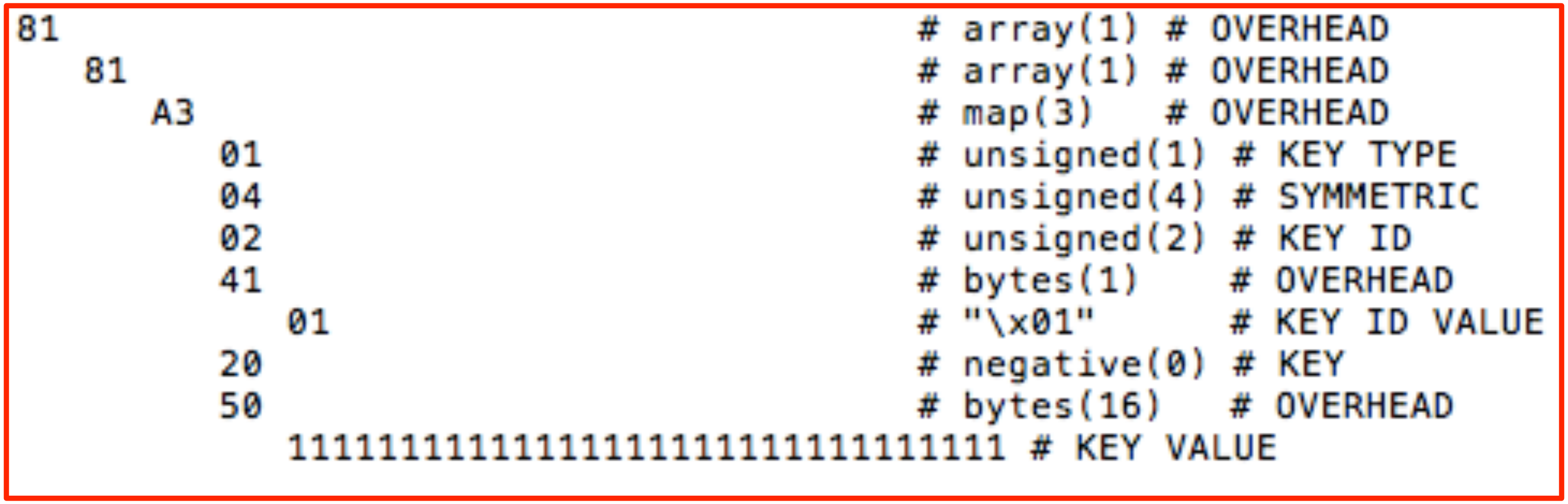
EUI-64 of Pledge

26 + 1 + 1 + 8

Content-Format removed

Issue #1: Packet size

- Join Response without short and JRC’s address



```
81                                      # array(1) # OVERHEAD
   81                                   # array(1) # OVERHEAD
      A3                                # map(3)   # OVERHEAD
         01                             # unsigned(1) # KEY TYPE
         04                             # unsigned(4) # SYMMETRIC
         02                             # unsigned(2) # KEY ID
         41                             # bytes(1)    # OVERHEAD
            01                          # "\x01"      # KEY ID VALUE
         20                             # negative(0) # KEY
         50                             # bytes(16)   # OVERHEAD
            11111111111111111111111111111111 # KEY VALUE
```

- 26 bytes to encode key (16 bytes) and key ID (1 byte)
- Uses CBOR + COSE structures
- Can be optimized with compressed COSE approach like used in OSCOAP

Issue #2: Join Proxy Policy

- Proposal: Provide a mechanism to accept insecure L2 packets at JP only upon a trigger (i.e. DAG root button press)

- Needed signal that join is allowed in EB
  - One option to use draft-richardson-6tisch-join-enhanced-beacon
  - Another option to reserve 0xFF of Join Metric in EBs to signal that node will NOT accept insecure L2 frames
    - Upon a trigger, fill Join Metric with the value according to RFC8180 (calculated from the DAG rank)

Conclusion

- PSK variant stable and implementation ready
- Settled down for EDHOC roles in the asymmetric variant, yet to implement
- Implementations of PSK variant available
- Will publish -04 with implementation experience before WGLC
- Reviews welcome