Minimal security framework for 6TiSCH

draft-vucinic-6tisch-minimal-security-00

October 28th, 2016

Mališa Vučinić, Inria
Jonathan Simon, Linear Technology
Kris Pister, UC Berkeley

Context

- Terminology
  - **JN**: Joining Node
  - **JCE**: Join coordinating entity
  - **JA**: Join assistant - radio neighbor of JN
- JN provisioned with a “join” credential
  - Pre-Shared Key (PSK)
  - raw public key (RPK)
  - Locally-valid certificate and a trust anchor
- Expects to be configured with
  - K2 from [ietf-6tisch-minimal]
  - short 802.15.4 address

Goals

- Minimize number of exchanges -> single round trip with PSKs

- Minimize join-specific code -> reuse of existing protocols

- Security -> end-to-end AES-CCM

Join protocol

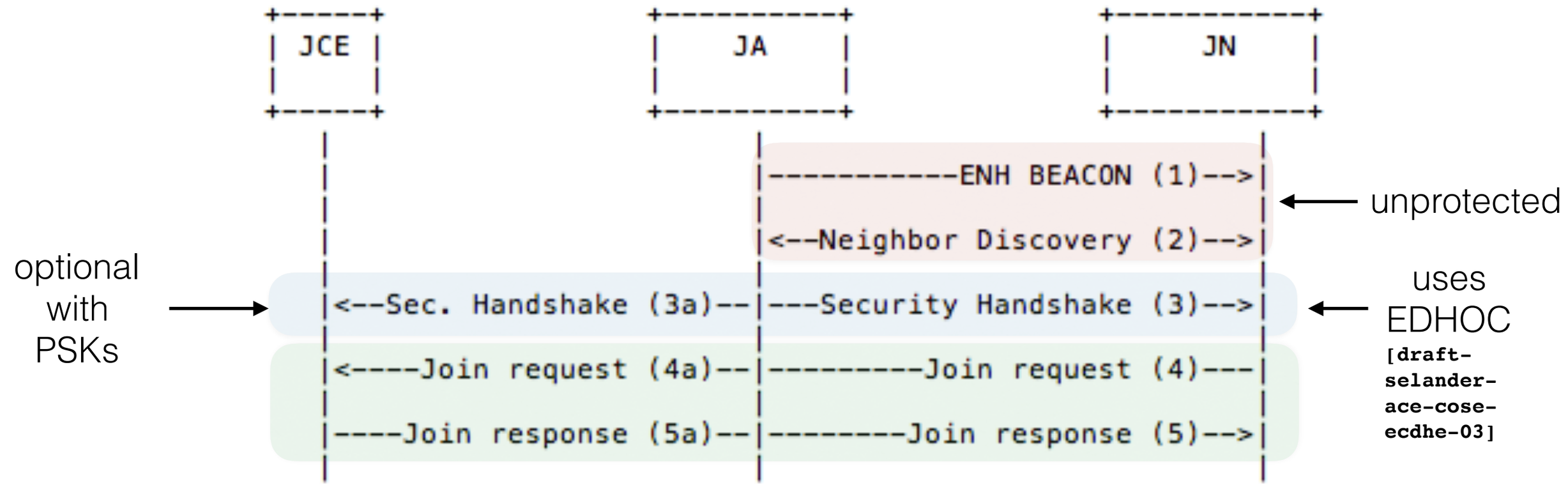
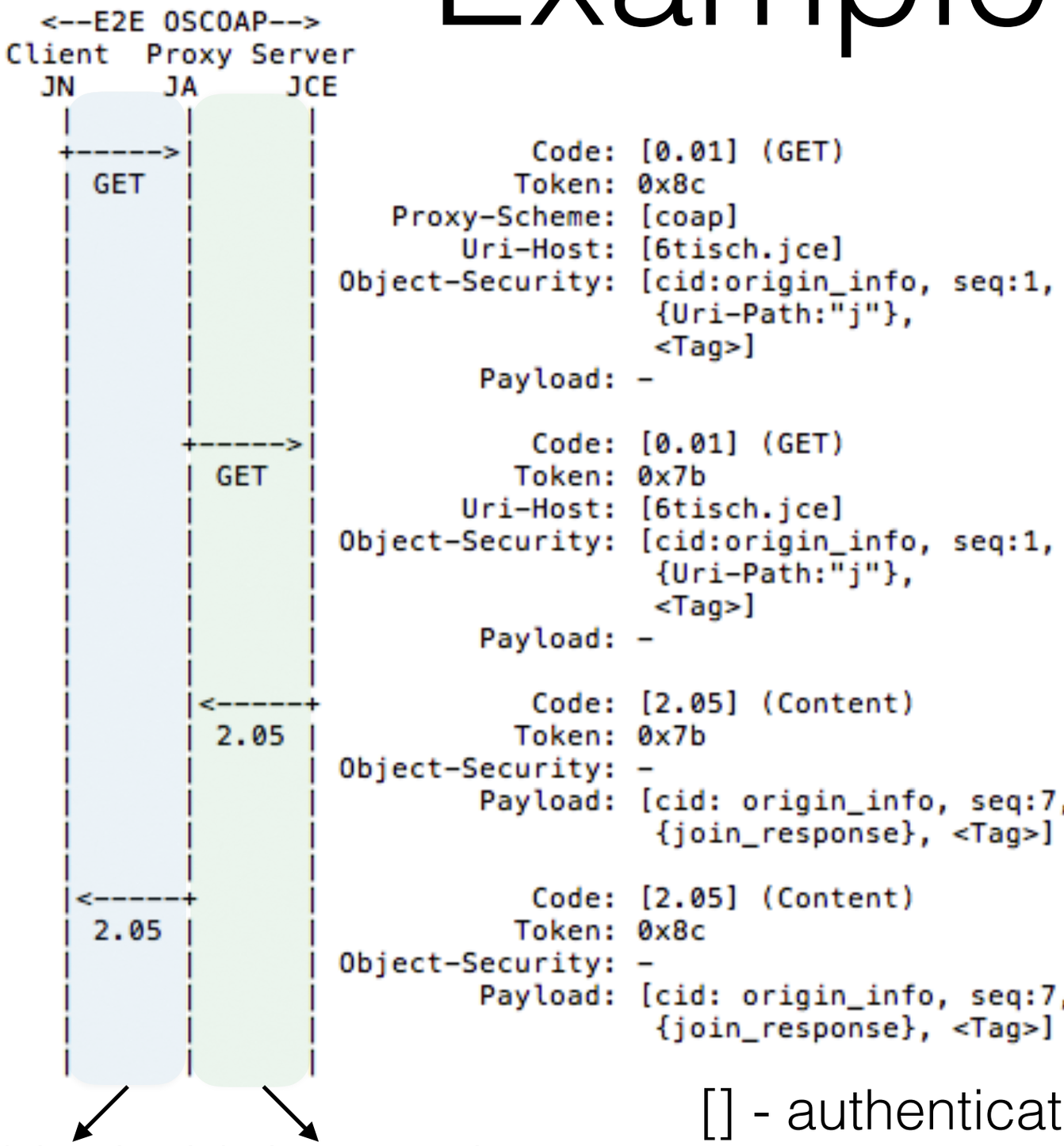


Figure 1: Message sequence for join protocol.

Protocol Specification

- Implemented with CoAP
  - JN is a CoAP client, JCE a server
  - JA is a CoAP proxy
    - Stateless using app-level info
- E2E encryption *through JA* using OSCOAP + COSE
- Actual “traffic keys” and nonces are derived from PSK

Example (PSK)




```
origin_info:

[
     h'00170d00060d9f0e', / JN's EUI64 /
     49152, / JN's UDP source port /
     0x8c   / JN's CoAP token /
]
```

Encodes to 15 bytes

```
join_response:
[
    [   / COSE Key Set array with a single key /
        {
            1:4, / key type symmetric /
            -1:h'e6bf4287c2d7618d6a9687445ffd33e6' / key value /
        }
    ],
    h'af93' / assigned short address /
]
```

Encodes to 26 bytes

[] - authenticated

{} - encrypted