Update on BRSKI-CLE: A Certificateless Enrollment Framework in BRSKI

draft-yan-anima-brski-cle-01

Lei YAN

IETF 118, November 2023
Issues received in the last meeting

- Issue 1 (from Steffen Fries): The cryptographic approach should be discussed with CFRG.
- Issue 2 (from Michael Richardson): COSE objects and ACE-EST should be compared with.
Issue 1: The cryptographic approach should be discussed with CFRG.

- All the mathematical algorithm is deleted from the draft.
- The draft is changed to an enrollment framework based on Key Encapsulation Mechanism (KEM).
  - Considering the evolution towards quantum-safe algorithms
  - KEM-based authentication is lightweight than signature-based authentication
    - KEM-based authentication resulted in a speed increase of 25 ms, a saving of 71% compared with signature-based authentication \cite{1}.

Issue 2: COSE objects and ACE-EST should be compared.

- The draft does not specify any **local credentials** any more.
  - This framework can issue:
    - Any **lightweight credentials**, such as CBOR Web Tokens (CWTs)
    - Any **credential references**
- The **use case** is clarified and detailed.
  - The CBOR encoded certificate chain is still heavy for the Class 1 constrained IoT devices (defined in RFC7228).
- All existing **authentication protocols** supporting the **KEM** mechanism are compared with.
  - EDHOC (used by ACE-EST)
  - IPsec
  - TLS
Use case

- The access gateway is required to **authenticate** every connected IoT device in the hospital.
  - Preventing **medical data theft**

- Medical Constrained IoT devices:
  - RAM for authentication < **10 KB**
  - Total RAM = **8 KB** in extreme condition

- This kind of constrained IoT devices are also common in scenarios other than in the hospital.
  - **Class 1** constrained devices: ~ **10 KB** RAM (RFC7228)

Medical Data Theft Security Incident in hospital

Examples of medical constrained IoT devices
Motivation

• The limited RAM resources make the Class 1 constrained IoT devices hard to use certificates.
• The CBOR encoded certificate chain is still heavy for the Class 1 constrained IoT devices.
  – The CBOR encoded certificate chain\(^1\):
    o 4 length: \(\sim\) 4 KB
    o 2 length: \(\sim\) 1.5 KB.
• All existing enrollment protocols of BRSKI are based on certificates.
• This draft propose a certificateless enrollment framework for constrained IoT devices.

\(^1\) I-D.ietf-cose-cbor-encoded-cert: "CBOR Encoded X.509 Certificates (C509 Certificates)"
Whose public key is used for Encapsulating in KEM:

**client end VS server end**

- **Client end:**
  - A unique public key is required to be configured on every IoT device.
  - Less efficient in deployment when the amount of IoT devices is huge.
  - EDHOC (I-D.ietf-lake-edhoc) and IPsec (RFC 9370)

- **Server end:**
  - Only one public key needs to be configured on the server end for dealing with an enormous amount of client ends (the IoT devices).
  - More efficient in deployment
  - **This draft** and TLS (I-D.wiggers-tls-authkem-psk and I-D.celi-wiggers-tls-authkem)
    - The client end is assumed to have previously known the server end's public key in [I-D.wiggers-tls-authkem-psk].
      - In the BRSKI scenario, a pledge cannot previously know a domain server's public key.
    - The client uses the certificate chain to authenticate the server in [I-D.celi-wiggers-tls-authkem].
      - As BRSKI has already built trust between the pledge and the domain before enrollment, using public key is enough.
Another change

- EDHOC is used for the mutual authentication between the pledge and the registrar in BRSKI, as shown in [I-D.ietf-lake-authz].
  - The pledge's credential is supported transporting by reference rather than by value.
- A constrained IoT device does not need to configure a public key to identify itself for the whole bootstrapping process.

![Architecture Overview](image)

![Transporting Credential by reference](image)
Basic protocol flow

[Diagram showing a protocol flow with steps A to E, indicating messages protected using AC's public key with [] and messages protected using a symmetric key with <>]

[] Indicates messages protected using AC's public key.
<> Indicates messages protected using a symmetric key.
Thank you!
Looking for co-authors!

Questions?
It is welcome to make comments in the email list.