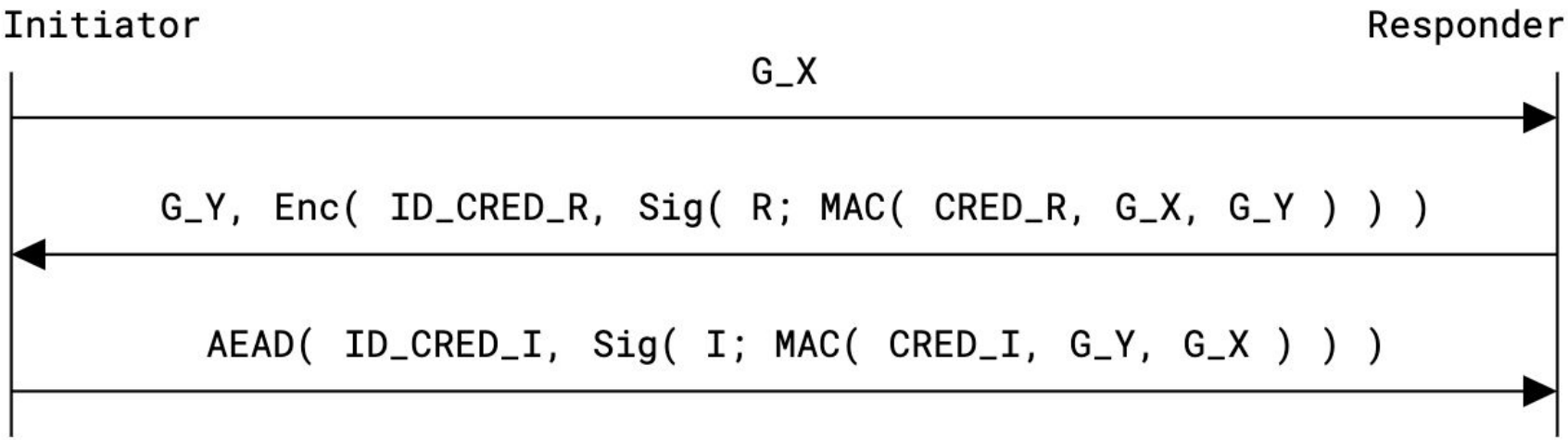
IETF 118
Ephemeral Diffie-Hellman over COSE implementation in Rust

https://www.ietf.org/archive/id/draft-ietf-lake-edhoc-22.html
https://github.com/openwsn-berkeley/edhoc-rs

CORE, ACE, LAKE table

EDHOC: Ephemeral Diffie-Hellman over COSE



minimal handshake = 101 bytes

https://www.ietf.org/archive/id/draft-ietf-lake-edhoc-22.html

edhoc-rs[1]

- A microcontroller-optimized implementation of EDHOC in Rust
  - no_std, no heap, inline CBOR encoding
- Effort towards formal verification with hax[2]
- Configurable crypto backends
- Skeleton for EAD handlers (extensions)

[1] https://github.com/openwsn-berkeley/edhoc-rs
[2] https://github.com/hacspec/hax

On the hackathon:

- ✅ improve processing, following implementation guidelines draft
  - support by-value or by-ref credentials
  - support EAD-aided credential validation
  - improve EAD handling
- ⏳ enable cryptographic backends as trait
- ⏳ model message flow as typestates
- ⏳ run edhoc-rs alongside a CoAP server in RIOT OS
- [ ] have a demo of lake-authz in the nRFs

TEAM:

- Christian Amsüss
- Geovane Fedrecheski
- Göran Selander
- Mališa Vučinić
- Marco Tiloca