FROST Messages & Data Serialization

April 15, 2021

Contents

1 Overview

The following document describes the byte-level structure of messages sent in FROST [1]. Each message consists of a fixed-sized header followed by the actual payload of the message.

2 Headers

All messages have the following header:

Bytes	Field Name	Data Type
1	`Message Type`	`u8`
1	`Version`	`u8`
2	`Sender ID`	`u16`
2	`Receiver ID`	`u16`

The `Message Type` and `Version` fields specify the payload that follows after the header. The sender is uniquely identified by the `Sender ID` [1] field and the receiver is uniquely identified by the `Receiver ID` [1] field.

3 Payload types

Messages in FROST are split into four general domains. The following sections describe each domain and its messages.

3.1 Key Generation with DKG

These messages are sent during the Distributed Key Generation (DKG).

3.1.1 Round One

Broadcast the public commitment vector $\vec{C} = \langle \phi_0, \ldots, \phi_{t-1} \rangle$ and the proof of knowledge $\sigma = (R, \mu)$.

Header:
`Message Type = 1`
`Version = 1`

Payload:

Bytes	Description	Data Type
2	Length `t` of the commitment vector	`u16`
$512 \cdot$ `t`	Individual commitments ϕ_j	`[AffinePoint; t]`
256	The value R	`Scalar`
256	The value μ	`Scalar`

3.1.2 Round Two

Broadcast the secret shares $f(l)$.

[1] TODO: Consider other data types such as `u32` or `u64`.

Header:
`Message Type = 2`
`Version = 1`

Payload:

Bytes	Description	Data Type
256	Secret share $f(l)$	`Scalar`

3.2 Key Generation with Dealer

...

Header:
`Message Type = 3`
`Version = 1`

Payload:

Bytes	Description	Data Type
...	...	...

3.3 Signing

...

3.3.1 Round One

Share signing commitments.

Header:
`Message Type = 4`
`Version = 1`

Payload:

Bytes	Description	Data Type
2	Number of signing commitments `n`	`u16`
$1024 \cdot$ `n`	Signing commitments	`[SigningCommitment; n]`

3.3.2 Round Two

...

Header:
`Message Type = 5`
`Version = 1`

Payload:

Bytes	Description	Data Type
...	...	...

4 Data Types

4.1 `AffinePoint`

4.2 `Scalar`

4.3 `SigningCommitment`

References

[1] Chelsea Komlo and Ian Goldberg. Frost: Flexible round-optimized schnorr threshold signatures. Cryptology ePrint Archive, Report 2020/852, 2020. https://eprint.iacr.org/2020/852.