Farcaster

Bitcoin-Monero atomic swaps

github.com/farcaster-project

Outline

- Part 1: Is Bitcoin digital cash?
- Part 2: Why Monero?
- Part 3: Atomic swaps & Farcaster
- Part 4: Q&A

But first!

Thanks to the CCS donors for funding us! 🎉

(more on the Monero Community Crowdfunding System later)

Part 1: Digital Cash

What is digital cash?

1. **bearer instrument**

   physical possession determines ownership

2. **peer-to-peer**

   no need for a third party to facilitate/mediate

3. **permissionless**

   no permission from a third party required

4. **fungible**

   valid bearer instruments of the same nominal value retain this value in reality too: all notes are equal

5. **privacy preserving**

   privacy is not secrecy, but the power to selectively reveal oneself to the world

   only the parties involved in the exchange have to be aware of the exchange's occurrence

Is Bitcoin digital cash?

1. **bearer instrument** ✅
   a. if you hold your own keys, you own them
2. **Peer-to-peer** ✅
3. **Permissionless** ✅
   a. exchange is permissionless, but addresses and utxo can be blacklisted and flagged as "dirty"
   b. as long as your signed transaction can reach bitcoin's p2p layer
4. **limited** fungibility ❌
5. **not** privacy preserving ❌
   a. Bitcoin's ledger is fully transparent.

What the hell do we need a cash equivalent for in the 21st century anyway? 💵==🧻💩?

1. less harmful example 👶🔪

consumer profiling - membership cards at retailers like Migros and Coop

offers convenience, but also strengthens targeted advertising

undermines consumers' agency in purchasing decisions

typically siloed, so although your bank/retailer may share/sell this information with third parties, this may still qualify as selective disclosure - not necessarily a full loss of privacy

2. Dangerous example 💣

Chinese Social Credit System

Dangerous lightcone paths of Bitcoin’s future

fully transparent ledger: transactions are visible in plaintext

While current financial system is complementary to surveillance, it is still typically siloed.

Bitcoin’s plaintext transactions are globally surveillable - a risk that must be managed

TECH (HTTPS://WWW.CNBC.COM/TECHNOLOGY/)

Bitcoin sleuthing start-up Chainalysis doubles valuation to $2 billion with Benioff backing

PUBLISHED FRI, MAR 26 2021·9:30 AM EDT UPDATED FRI, MAR 26 2021·9:48 AM EDT

Monero

1. bearer instrument ✅

   if you hold your own keys, you own them

2. peer-to-peer ✅
3. permissionless ✅
4. fungible ✅
5. privacy preserving, up to plausible deniability ✅

I want privacy for my transactions! do I kraken btc/xmr now?

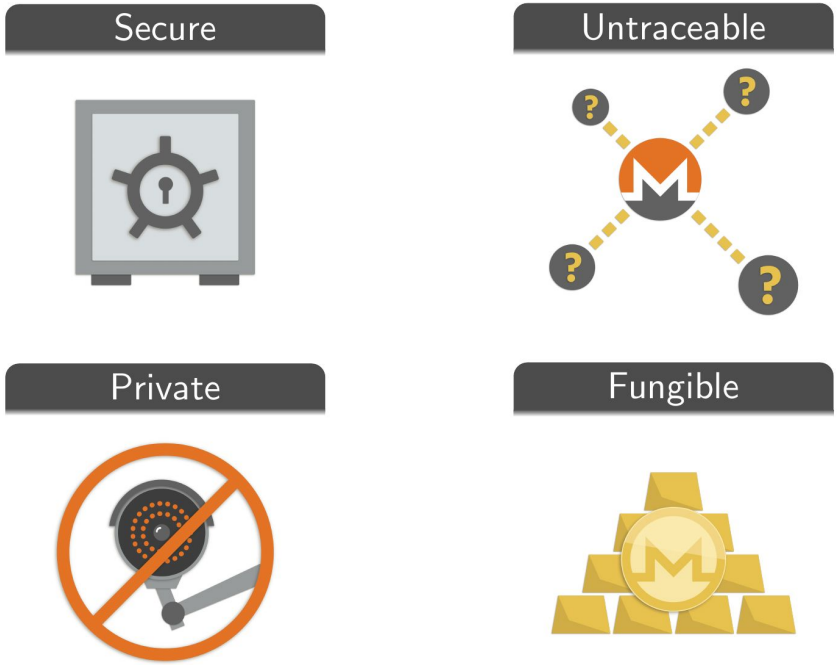
- going through a centralized exchange still requires trust, and leaks your data
- cash should be acquired via a peer-to-peer cash exchange!
- Farcaster is a protocol for executing atomic swaps that will implement exactly that

https://github.com/farcaster-project/

Part 2: Why Monero?

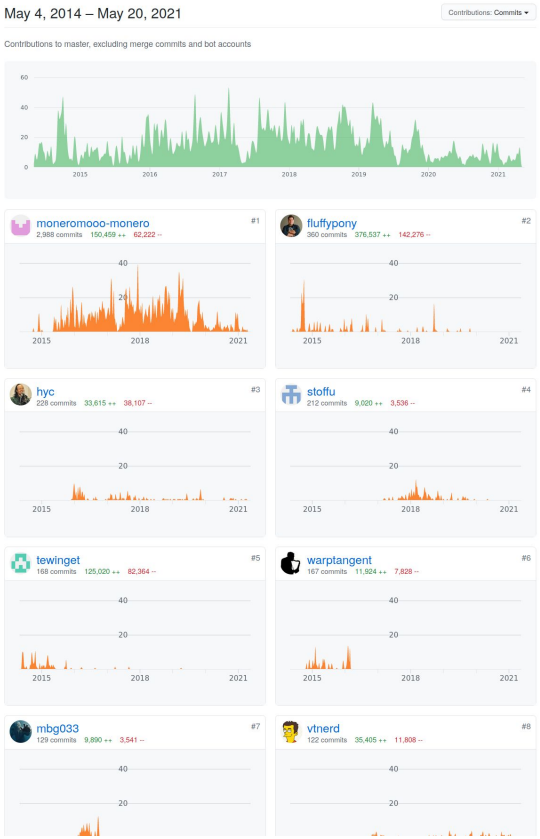
Why Monero?

- p2p cash



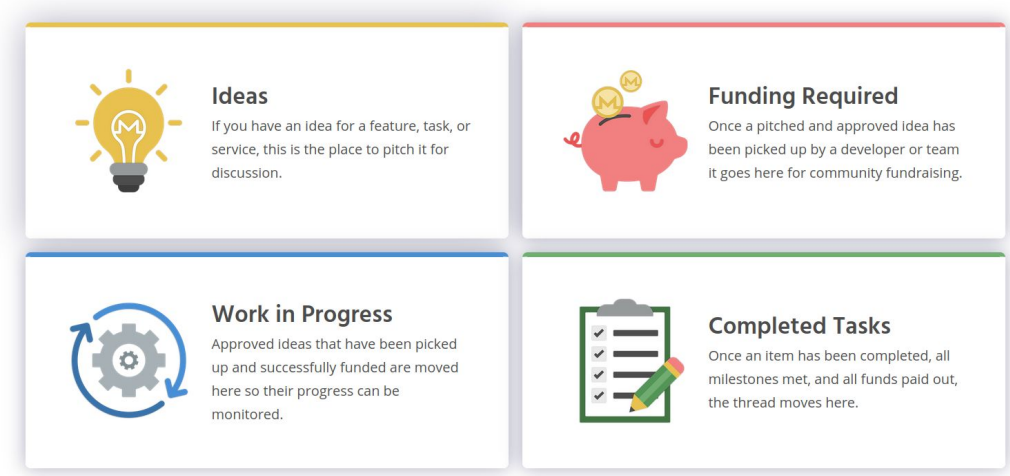
Why Monero?

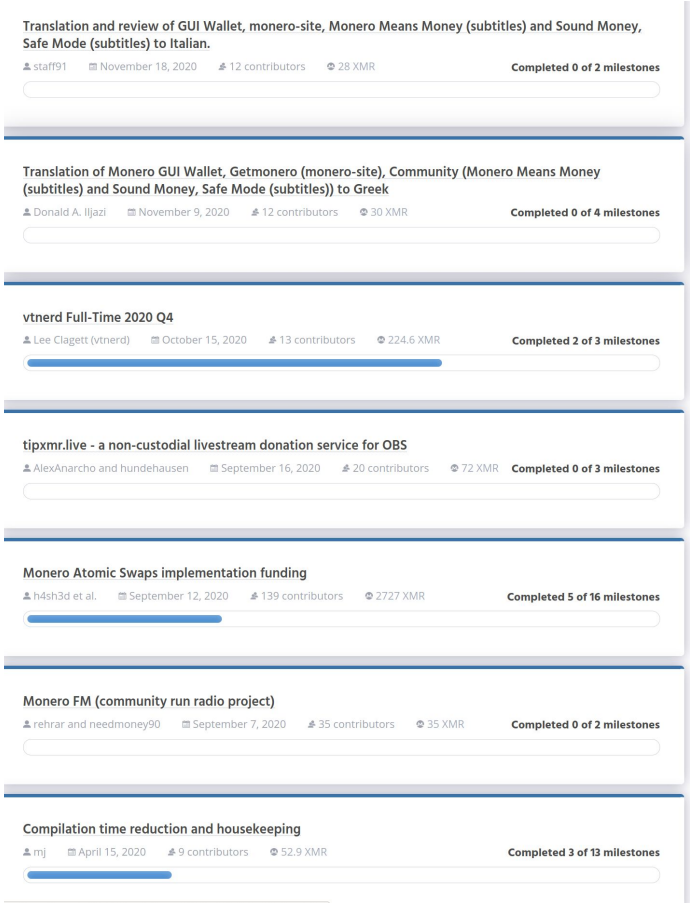
- Decentralized, actually used, high dev activity



Why Monero?

- Community Crowdfunding System





Why Monero?

- p2p cash
- Decentralized, actually used, high dev activity
- Community Crowdfunding System

DON'T BUY

MONERO

Cryptocurrencies are harmful to the banking system and may weaken the state apparatus

Monero Features

- Ring signatures, confidential transactions, stealth addresses
- Tor / I2P support
- Dandelion + noise
- ASIC-hard proof of work
- No supply cap, but tail inflation (0.6 XMR mining subsidy forever)
- Reproducible builds

Monero Address Description

- Concatenation of network byte + public view key + public spend key + checksum

4AdUndXHHZ6cfufTMvppY6JwXNouMBzSkbLYfpAV5Usx3skxNgYeYTRj5UzqtReoS44qo9mtmXCqY45DJ852K5Jv2684Rge

- Private view key discovers transaction history/balance

- Private spend key signs transactions

Transaction Format Comparison

BLOCKCHAIN Products Data Explorer Login Sign Up

BTC / Transaction USD BTC

View information about a Bitcoin transaction

Summary

Hash	e60c55312e3cbce1798c0cea08f1ee7be39c1204286ff541511e...			2019-11-20 21:31
	1BZWUxLcWGPvvwkJEtyraQX8jan8Vi3abB	0.59726198 BTC	1J9eHaNdbnJwcdWJnJRimACr8EXDct18Y3	0.11322400 BTC
			1BZWUxLcWGPvvwkJEtyraQX8jan8Vi3abB	0.48290798 BTC
Fee	0.00113000 BTC (502.222 sat/B - 125.556 sat/WU - 225 bytes)		1 Confirmations	0.59613198 BTC

Transaction Format Comparison

Confirmations	3
From Block	2364877
Output total	confidential
Fee	0.000009860000 XMR
Size	1453 bytes
Mixin	10
Unlock	0

Inputs (1)

Amount	Key Image
0.000000000000	eb507d5435c90dadece47da19b2d0bf67be9fde8bc16f1c74d53a7937bfa085d

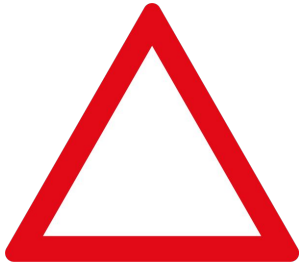
From Block	Public Key
2257511	d85f170d54655ab35ea5a30bf8e9323339abea1c3d725ede6b6653dd73245fe3
2351797	78d50990e52bea25aa844236adf48d8e48a58d7b521e1c5e86124438370fb132
2363542	2da4623b67d6b5858226ad2e45ab3f21b09f867cffc45a81cedb14c7befcf955
2363552	503c935af50ec8787589061e7bad81a3ec62006482b0660c1d68e94aa27c6e98
2363961	72a74ca3a2572c957023cffd0bdce11e37fe173cc20ce3d07edf114c9ef28537
2364033	8e94657a68116e7330ca9f936719236e1ff6690fe6dcca4b4d6ab992ba4a240d
2364512	925caf150613f2663f1558c3c229cc637ec02ace5646a22d31121e2f58846e97
2364549	ef8da0c998598a365d0869b21c9027126da16cc35b4aeb2c98d30e8b60811253
2364696	8a895a277ca242173f9562b10ffbd66d212bd25d101f4fa6b30b0dddbd2c374f
2364739	c5068ada875493e2cb9c2701072431b5bb286dbaf1aa78f7a16f5b83293a8269
2364797	00badbdc2d77ebe6d3f1b17766dc0714c6fe2a0bd26bd1d8555c0ce904a94c53

Outputs (2)

Amount	Public Key
0.000000000000	545c256981a145b7db477f01ee4679692da673cc1d37f60cd8f7392919f1466c
0.000000000000	26fbf842e14cf801f905cab2c8716c8ae9fe4f525e053b71e8ee267d7d42f1cc

Downsides to Monero features

- Stealth Addresses
  - Need for scanning all transactions
- Confidential Transactions
  - Larger tx size
  - longer tx verification
- Ring Signatures
  - No transaction chaining
  - No UTXO set



Monero atomic swap challenges

- No scripting capability
  - Multi-signature possible off-chain
- Weird timelocks (not what we want)
  - Monero timelock locks all outputs indiscriminately
- No SegWit-like unbroadcasted transaction chains
  - Chaining unmined transactions impossible

To the rescue!

Cryptology ePrint Archive: Report 2020/1126

Bitcoin-Monero Cross-chain Atomic Swap

Joël Gugger

Part 3: Atomic swaps & Farcaster

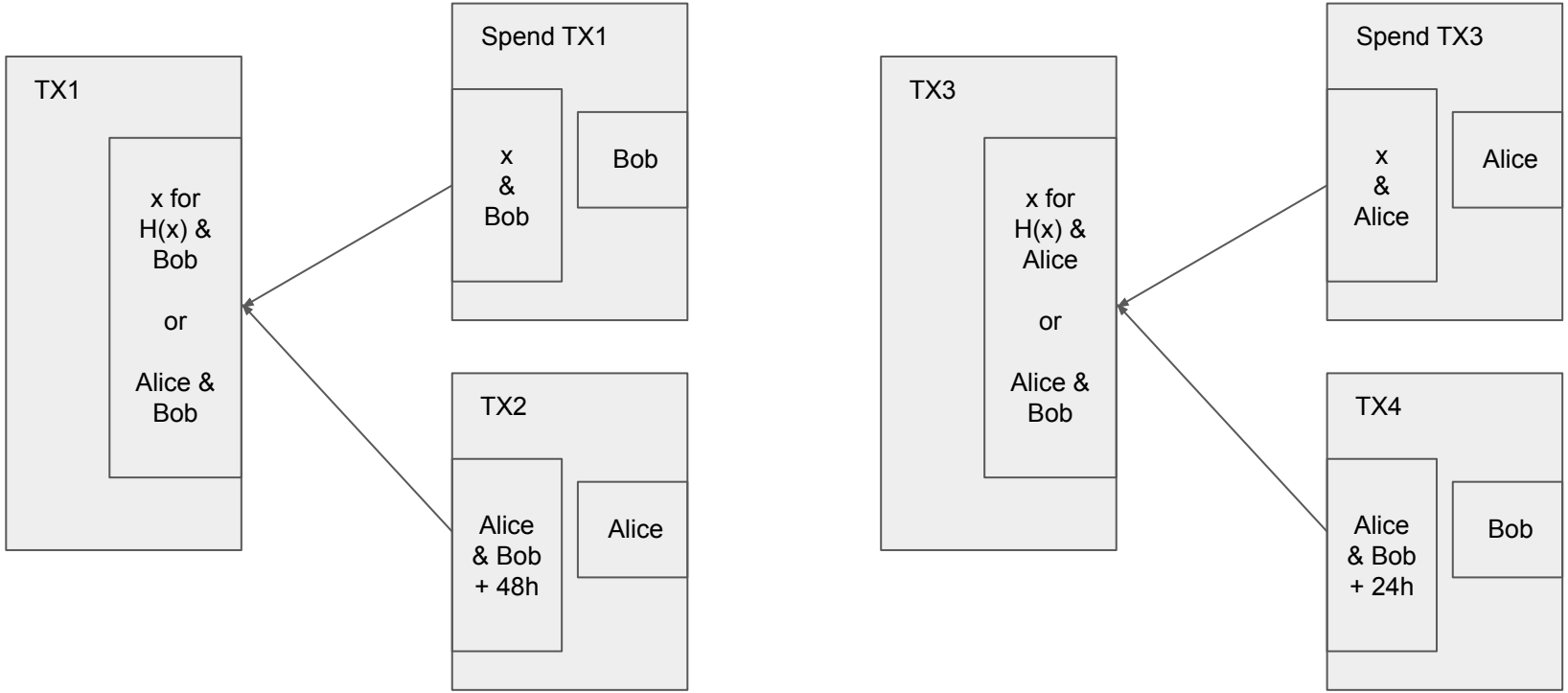
Part 3

- HTLC based atomic swaps
  - UTXO-based transactions structure
  - Example of a “standard” protocol
- Adaptor signatures
  - Concept and general overview
- Farcaster
  - Features
  - Protocol walkthrough
- Taproot
  - How to improve Farcaster

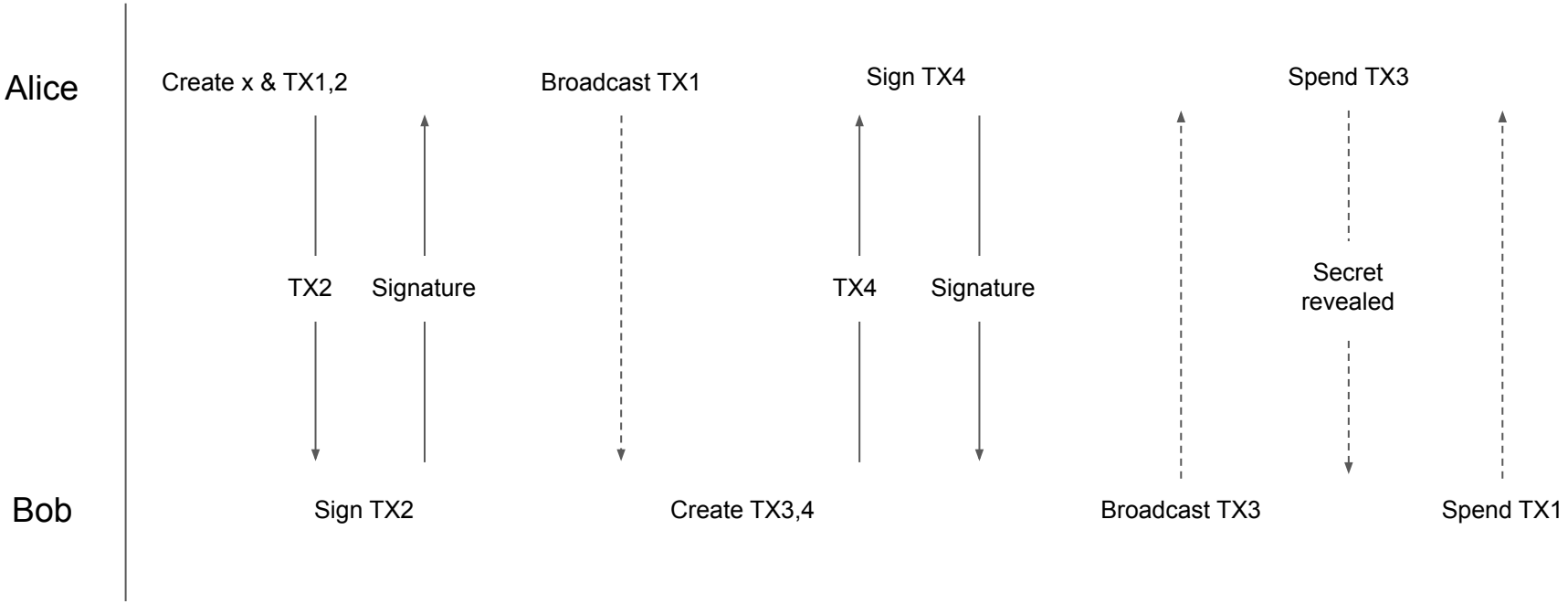
Cross-chain Atomic Swaps

- Exchange of two blockchain coins,
- in an adversarial environment,
- guaranteed atomic if the protocol is followed;
- i.e. either the trade succeed,
- or is reverted.

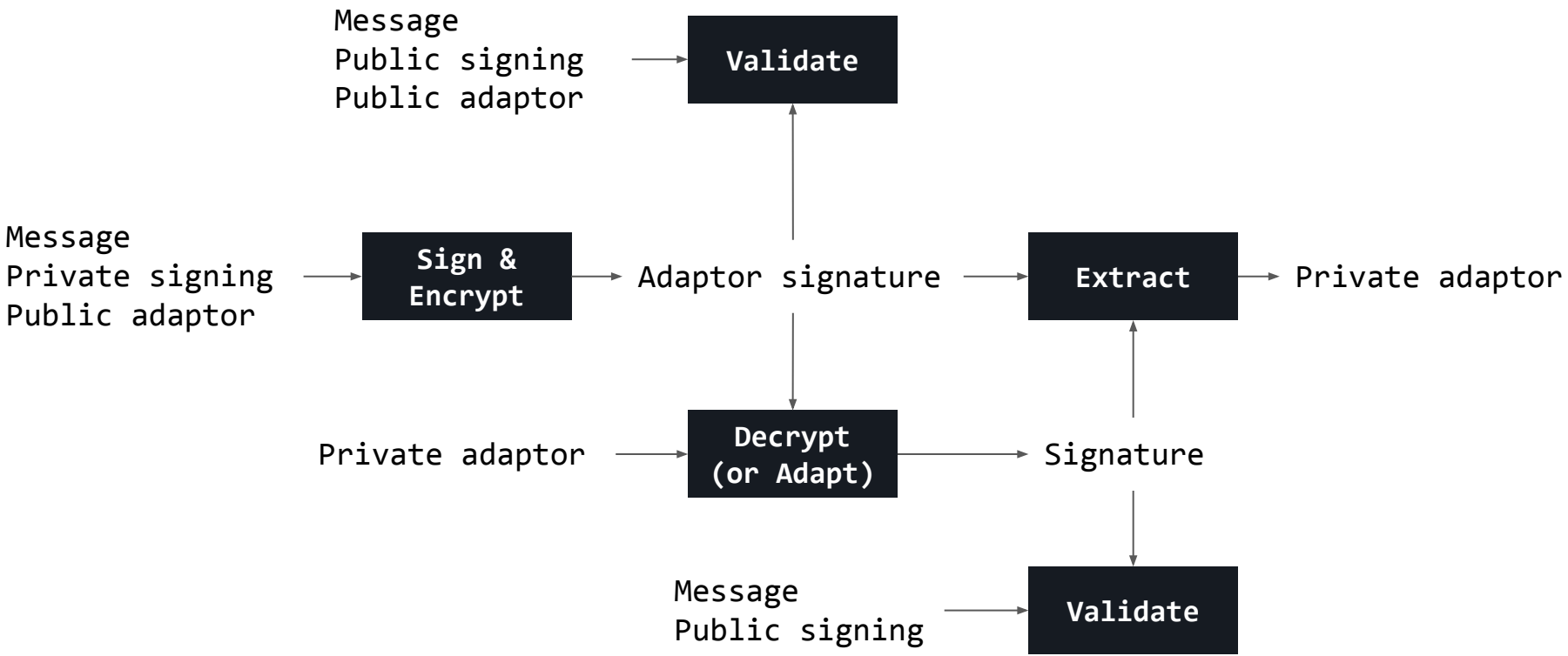
Hash Time Locked Contract



Hash Time Locked Contract



Adaptor signature (or one-time VES)



Farcaster features

- Treats blockchains differently based on their on-chain features
  - Define two blockchain roles: “Arbitrating” and “Accordant”
- One blockchain (the “Arbitrating”) needs:
  - Timelocks
  - “Scripts”
  - Equivalence of SegWit in Bitcoin for UTXO-based blockchains
- The other (the “Accordant”) <u>doesn’t</u>

Idea

Create a 2-of-2 “multisig” (not really) on the accordant blockchain.

Lock accordant coins inside.

Sell (reveal) one of the two shares on the arbitrating blockchain using an adaptor signature.

```
k = k^a + k^b (mod ...)

where
    k^a: Alice's private key;
    k^b: Bob's private key;
    k: full private key;


k^b = k - k^a (mod ...)
```

Problems

How to ensure that a share is always revealed?

What if arbitrating and accordant elliptic curves are different?

```
e.g.:

bitcoin: secp256k1
monero: curve25519
```

Always reveal

Managed with game theory by introducing a punishment mechanism, like Lightning Network channels.

```
The arbitrating refund
transaction that returns the
accordant coins to its
original owner may be punished
if not broadcasted on time.
```

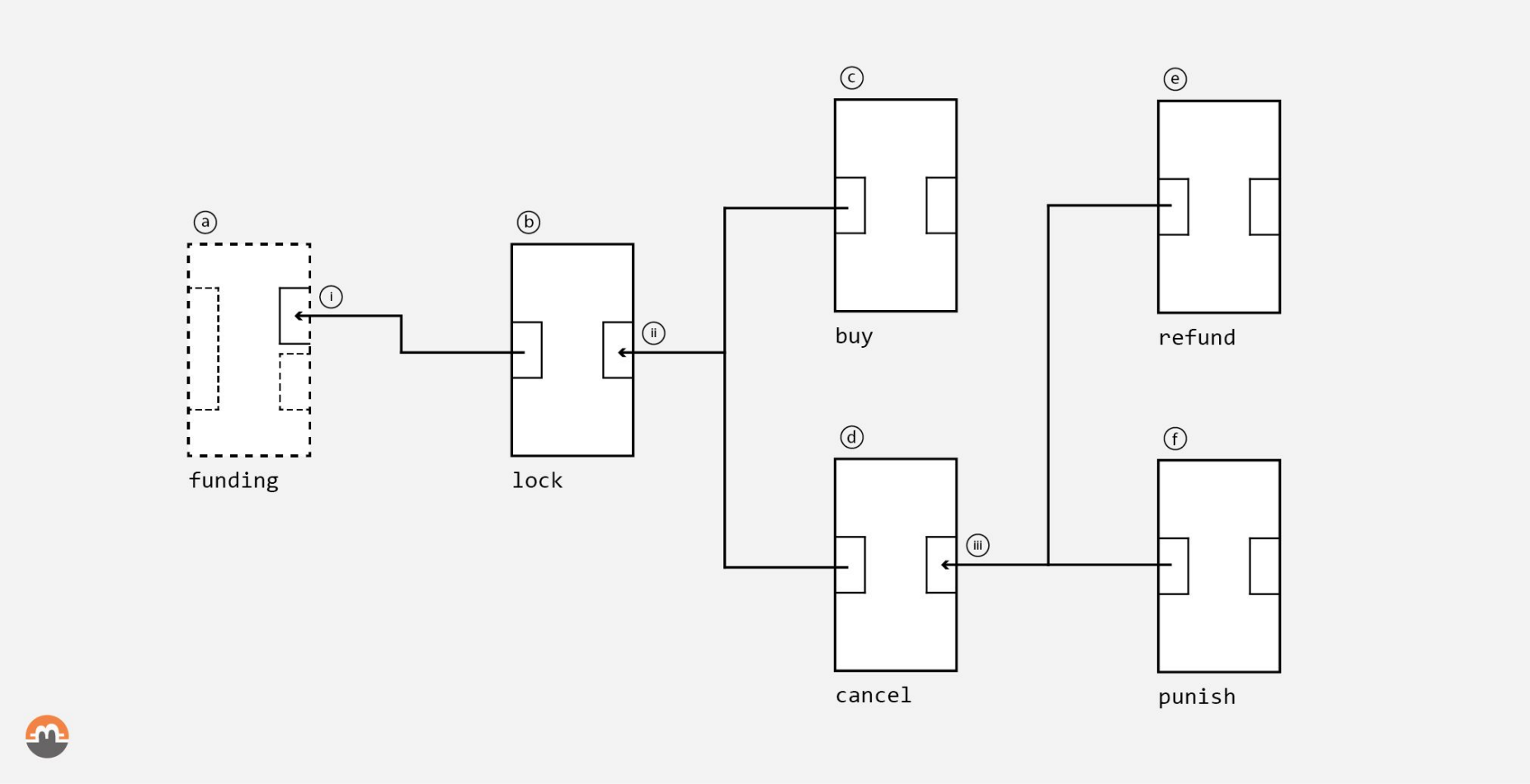
Different elliptic curves

Cross-group discrete logarithm equality proof in zero-knowledge.

Prove that it exists a relation between two points over two prime-order groups where the discrete logarithm problem is assumed to be hard.

Each participant generate one *x* and its proof.

```
select random x < min(p, l)

t = project x over secp256k1
T = tG

k = project x over curve25519
K = xH

Prove(T, K, G, H)
```



Source: https://github.com/farcaster-project/RFCs

lock (ii):

```
IF
    2 <Alice's Ab PubKey> <Bob's Bb(Ta) PubKey> 2 CHECKMULTISIG
ELSE
    <num> [TIMEOUTOP] DROP
    2 <Alice's Ac PubKey> <Bob's Bc PubKey> 2 CHECKMULTISIG
ENDIF

where
    Ab: Alice's buy key;
    Bb: Bob's buy key;
    Ac: Alice's cancel key;
    Bc: Bob's cancel key; and
    Ta: Alice's adaptor key
```

buy:

```
0 <Bob's Bb(Ta) signature> <Alice's Ab signature> TRUE <script>
```

cancel:

```
0 <Bob's Bc signature> <Alice's Ac signature> FALSE <script>
```

cancel (iii):

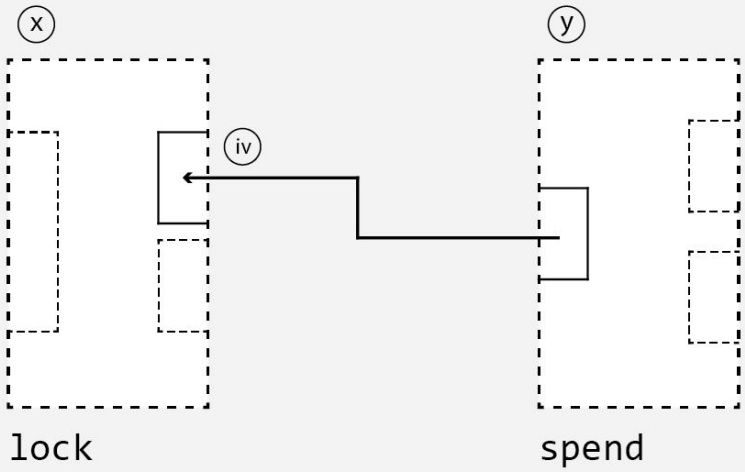
```
IF
    2 <Alice's Ar(Tb) PubKey> <Bob's Br PubKey> 2 CHECKMULTISIG
ELSE
    <num> [TIMEOUTOP] DROP
    <Alice's Ap PubKey> CHECKSIG
ENDIF

where
    Ar: Alice's refund key;
    Br: Bob's refund key;
    Ap: Alice's punish key; and
    Tb: Bob's adaptor key
```

refund:

```
0 <Bob's Br signature> <Alice's Ar(Tb) signature> FALSE <script>
```

cancel:

```
<Alice's Ap signature> FALSE <script>
```

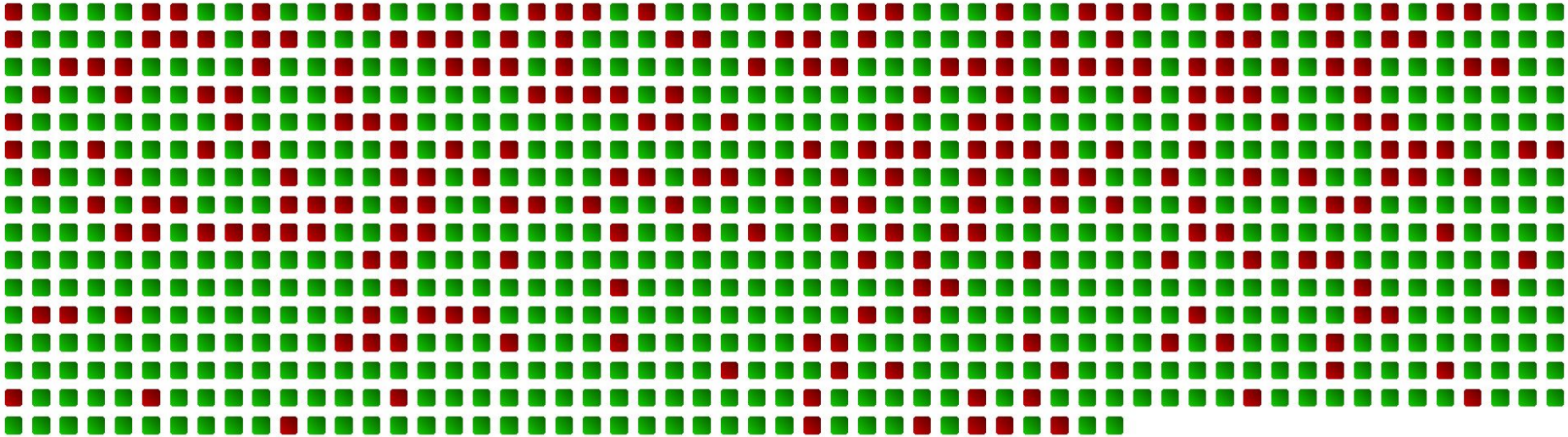


Source: https://github.com/farcaster-project/RFCs

Protocol

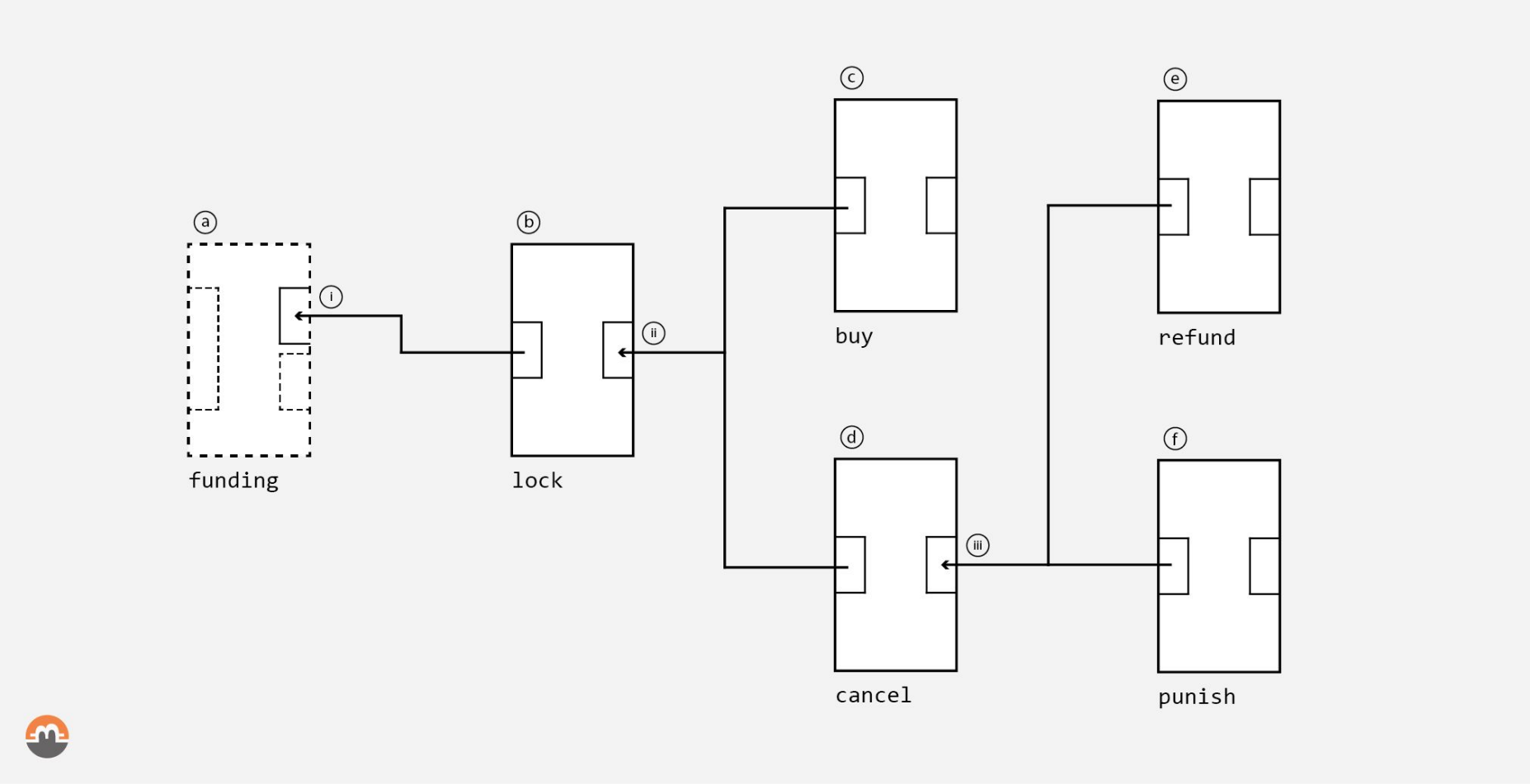
- Initialize the keys, exchanges the public parts, share the proofs and share the private Monero view key
- Create the Bitcoin core transactions (lock, cancel & refund)
- Co-sign the cancel and prepare the adaptor signatures
- Alice share her adaptor signature (refund)
- Bob lock the bitcoin (on-chain)
- Alice lock the monero (on-chain)
- Bob share his adaptor signature to Alice (buy)
- Alice adapt Bob’s signature and take the bitcoin, revealing her key
- Bob compute the full Monero private spend key

#Taproot



05/20/2021 4:30 pm

Source: https://taproot.watch



Source: https://github.com/farcaster-project/RFCs

lock (ii) Taproot with scripts:

```
        Q                       | the Taproot tweaked key
        |
---------------------
|                   |
P           Script Merkle root  | P: an internal key with
                    |               unknown DLog
                    |
    /-------------------------\
TapLeaf buy script      TapLeaf cancel script   | SegWit v1
                                                  scripts
```

lock (ii) Taproot with scripts:

```
<Alice's Ab PubKey> CHECKSIG <Bob's Bb(Ta) PubKey> CHECKSIGADD m
NUMEQUAL

where
    Ab: Alice's buy key;
    Bb: Bob's buy key; and
    Ta: Alice's adaptor key
```

buy:

```
<nitems> <len> <input> <len> <script> <len> <c>

where
    <input>: <Bob's Bb(Ta) signature> <Alice's Ab signature>
    <script>: TapLeaf buy script
    <c>: control block
```

lock (ii) Taproot with scripts:

```
<num> [TIMEOUTOP] EQUALVERIFY DROP
<Alice's Ac PubKey> CHECKSIG <Bob's Bc PubKey> CHECKSIGADD m
NUMEQUAL

where
    Ac: Alice's cancel key; and
    Bc: Bob's cancel key;
```

cancel:

```
<nitems> <len> <input> <len> <script> <len> <c>

where
    <input>: <Bob's Bc signature> <Alice's Ac signature>
    <script>: TapLeaf cancel script
    <c>: control block
```

lock (ii) Taproot with MuSig2:

```
            Q                        | the Taproot tweaked key
            |
    ----------------------
    |                    |
    P           Script Merkle root   | P: the internal key, a MuSig2
                         |               setup
                         |
            TapLeaf cancel script    | SegWit v1 script
```

```
P: Ab + Bb + Ta

where
    Ab: Alice's buy key;
    Bb: Bob's buy key; and
    Ta: Alice's adaptor key
```

What’s next?

- Taproot, privacy++
- MuSig2, privacy+++
- Channels! Channels everywhere! Speed swaps

Q&A

Resources

Farcaster GitHub: https://github.com/farcaster-project

#monero-swap on freenode -- soon on libera.chat ;)
- Weekly meeting on Wednesday at 16:00 UTC

Our ongoing CCS project:
https://ccs.getmonero.org/proposals/h4sh3d-atomic-swap-implementation.html