Secure Asset Transfer Protocol (SATP)

Chair Document Review

Are the documents ready to go?

Agenda:

1. The Good News

2. The Bad News

The Good News

- The documents have come **so so so so so** far!
- To the authors and WG participants: very well done

The Bad News

- Your bad cop chair did a full nit-pick review
  - Can “will two independent implementations run into problems?”

- A bunch of small nit problems
  - I will write these up and send them separately

- Two larger problems
  - Encoding formats
  - List of acceptable values

Encoding formats

- TL;DR: values with non-obvious datatypes need to indicate an encoding format

Example from the document:

```
* originatorPubkey REQUIRED.  This is the public key of the asset
  owner (originator) in the origin network or system.

"originatorPubkey":
  "0304b9f34d3898b27f85b3d88fa069a879abe14db5060dde466dd1e4a31ff75e44",
```

- Good: it has an identifying name
- Good: it indicates OPTIONAL vs REQUIRED
- Good: there is an example!
- Unfortunate: keys can be encoded in many ways…
  - binary? Hex? Hex with 0x prefix? Base64? Base64 DER? Base64 CER? Base64 FP?

List of Acceptable Values Needed

- TL;DR: interoperability typically requires an explicit list of acceptable values

Example from the document:

* networkLockType REQUIRED: The default locking mechanism used by a network. These can be (i) timelock, (ii) hashlock, (iii) hashtimelock, and so on (TBD).

```
"networkLockType": "HASH_TIME_LOCK",
```

- Good: it has an identifying name
- Good: it indicates OPTIONAL vs REQUIRED
- Good: there is an example!
- Unfortunate: the example doesn't match the text
- Unfortunate: what values are ***the exact*** values allowed?
  - HASH_TIME_LOCK? hashtimelock? TBD???

List of Acceptable Values: Options

- There are a few paths forward from here

- Options for **each** field that has a restricted list of values:
  - List all the current acceptable values
  - Point at an external list of acceptable values
  - Create an IANA registry table for values likely to be extended by multiple other future documents

Summary

To be fully implementable by someone that is coding in a box:

- You need to be very very precise

- All options must be carefully articulated

  - Optional / required: DONE!
  - Naming: DONE!
  - Flow and handling: DONE!
  - Error control: DONE!

  - Value Encoding: Inferred only by example
  - Value limitations: Missing