RWOT XI Advance Paper - DID:torrent

Date: July 19 2022 Author: Vinay Vasanji Version: 1.0

Context

On July 19 2022 Decentralized Identifiers (DIDs) v1.0 changed status to a W3C Recommendation, meaning it is now an official Web standard.

Specific Problem

There are currently >130 DID Methods, as tracked by the diddirectory.com. These specify the precise operations by which DIDs are created, resolved, updated, and deactivated (CRUD).

Verfiable Data Registries (VDFs) are the systems or substrates that facilitate CRUD operations for DIDs.

DID Methods rely on a 3rd party or parties for CRUD operations. For example, DID:3's VDR is the Ceramic Protocol or Network, which itself is dependent on IPFS and Ethereum. You are beholden to whomever secures the underlying VDR. The exception is DID:peer which describes a synchronization protocol between peers for the Verifiable Data Registry, but does not provide a precise description of CRUD operations.

Furthermore, despite the technical merits of DIDs, there is limited real world adoption in our digitally intermediated lives.

Specific Critique

In short, there are three issues: 1. Most DIDs are not peer-to-peer, therefore not truly decentralized. 2. DID:peer does not specify precise CRUD operations. 3. DIDs in general lack adoption, therefore have had limited positive impact so far.

In this Advance Paper we outline a novel approach to addressing the aforementioned issues.

Specific Solution - Part A

DID:torrent is a new DID Method that relies on BitTorrent, the worlds most popular and robust peer-to-peer protocol. DID:torrent uses two BitTorrent Enhancement Proposals (BEPs), namely BEP5 and BEP44, to faciliate CRUD operations: - **BEP5** introduced trackerless torrents, this decentralizes peer discovery as each peer becomes a tracker. Note: a bootstrap server is still required, but sufficient redundancy can be built into the DHT by enabling multiple, unaffiliated bootstrap servers. - **BEP44** is an extension that enables storing and retrieving arbitrary data in the BitTorrent DHT. It supports both storing immutable items, where the key is the SHA-1 hash of the data itself, and

mutable items, where the key is the public key of the key pair used to sign the data.

SHA-1 is considered unsafe by NIST. BEP52 introduces BitTorrent v2.0 which utilizes SHA2-256, however this improvement is likely irrelevant for this proposed DID:torrent implementation.

The rationale is we are only interested in the mutable items feature in BEP44. There is no arbitrary data required in the BitTorrent DHT, except for the Public Key of each network participant and an Info Hash.

To articulate more succinctly, the CRUD operations for DID:torrent are as follows:

- **CREATE**: Connect to the network via bootstrap server(s), announce Info Hash, and generate and store your public key in the DHT as a mutable item (via BEP44).
- **READ**: Connect to the network via bootstrap server(s), announce/retrieve Info Hash, and identify relevant peers' public keys.
- **UPDATE**: Connect to the network via bootstrap server(s), announce Info Hash, and store an alternate public key in the DHT as a mutable item (via BEP44).
- **DELETE**: Disconnect from the network, stop announcing/seeding an Info Hash. Disconnection removes your public key from the network.

This method creates a truly p2p VDF (Verifiable Data Registry) with a BitTorrent DHT. The question then is, what is the Info Hash in the context of this VDF/decentralized identity network.

Specific Solution - Part B

In Part A we specified a DID Method that utilizes the BitTorrent DHT, and enables the temporary storage and retrieval of a Public Key in the network by each peer. However, we require an 'application' to both encourage large scale adoption of the Method and provide the basis for the Info Hash that facilitates peer discovery.

For that, we have built a p2p chat app: candi.es. Participants join a channel, i.e. candi.es is a global channel, candi.es/rwot is local channel. Channels are user-generated, and each channel has a unique identifier. The webapp utilizes WebTorrent, a Node.js client library that enables BitTorrent DHT, including BEP44, to run directly in any browser with no additional client software or installation. This minimizes friction with participant onboarding.

Each channel identifier is converted to an Info Hash and announced by a peer into the DHT, which returns a list of other peers with that Info Hash and their Public Key. These web peers are other participants connected to a specific channel in our app.

Why Does This Matter?

- DID:torrent is a truly decentralized DID Method, offering an improvement on DID:peer, with specific CRUD operations.
- Candi.es is a web3 social application to bootstrap, and scale, the adoption of DID:torrent.
- Candi.es and DID:torrent together offer the first real world example of verb-like digital identity, rather than noun-like digital identity. Refer to Sheldrake, 2022.

Considerations for RWOT

DID:torrent and the Candi.es application will benefit from the following non-exhaustive debate and consideration, however additional critique and conversation is always encouraged: - Wallet architecture design for generating, and resolving, keys for DID:torrent. - Merits and/or risks associated with using BitTorrent key-formatting ed25519-supercop vs less obsure formatting. - Risk assessment of SHA-1 in the context of this specific implementation, i.e. DID:torrent plus Candi.es. - Evaluation of the Generative Identity principles against this specific implementation, i.e. DID:torrent plus Candi.es (note may need to develop an evaluation methodology, similar to the DID Method Rubric). - Sustainable funding model, to enable both the DID:method, and Candi.es to be digital public goods or infrastructure for web3. - What are the GDPR implications, if any, given Deletion (in DID:torrent's CRUD operations) involves simply disconnecting from the network. - Explore Schnorr, adapter signatures, FROST, applications for BEP44 mutable items in DID:torrent.

We invite potential collaborators to explore this further with us at RWOT.

Sincerely,

Vinay Vasanji