A system for distributed minting and management of persistent identifiers

Ł. Bolikowski, A. Nowiński, W. Sylwestrzak

ICM, University of Warsaw

IDCC 2015, London, 9-10 February 2015
Motivation and vision
Truly long-term persistence of identifiers should account for risks to managing organizations’ longevity.

- Internal risks: solvency, good will.
- External risks: political, economic, military.
Decentralize!

First steps already taken: w3id is based on a social contract between several managing organizations.

We can do even better (in terms of security, scalability, robustness), thanks to two great innovations in computer science: public-key cryptography and P2P networks.
Introducing Peer-Minted Persistent Identifiers (PMPIs), a distributed system in which anyone can mint and manage their IDs. The entire database with full history is public and stored in many copies (LOCKSS-like approach to long-term preservation).

If you know Bitcoin, keep this analogy in mind throughout this talk:

PMPI is to a traditional PID (backed by a minting org.)

as

Bitcoin is to a traditional currency (backed by a central bank)
Monetizing trust and QoS instead of being first.

Vision: the PMPI protocol as a common good, with multiple stakeholders building their business models around it.

Analogy: the Internet Protocol is a common good, yet we have for-profit and non-profit organizations offering their services on top of it (ISPs, domain names, hosting, SSL certificates).
Technical details
Anyone can mint an ID and associate with it a URL to content + a list of authorized keys.

For a given ID, any authorized key owner can alter the URL and the list of authorized keys.

Anyone can download the complete set of IDs with full revision history and verify its integrity.

No proper subset of participants is capable of shutting down the system.
An **operation** is the basic building block:

- mint a new identifier (set the URL to metadata + the list of authorized keys)
- update an identifier (change the URL to metadata + the list of authorized keys)

Operations form a **chain**, each operation is signed with an authorized private key, each includes the hash of the previous operation in the chain.

(Operations are immutable, tampering with the past is practically impossible)

Adding an operation to the chain requires a **proof-of-work**.

(Prevents spam and gives time to propagate operations across the network — we want a chain not a tree)
Chain of operations

1. Hash of the previous operation
2. Referenced ID
3. New ID record content
   - Resource location
   - Owner public key 1
   - Owner public key 2
   - ....
4. Digital signature of the transaction (above content) with owner's private key
5. Hash of the operation (above content)

Next step:

1. Hash of the previous operation
2. Referenced ID
3. New ID record content
   - Resource location
   - Owner public key 1
   - Owner public key 2
   - ....
4. Digital signature of the transaction (above content) with owner's private key
5. Hash of the operation (above content)

Final step:

1. Hash of the previous operation
2. Referenced ID
3. New ID record content
   - Resource location
   - Owner public key 1
   - Owner public key 2
   - ....
4. Digital signature of the transaction (above content) with owner's private key
5. Hash of the operation (above content)
For the sake of mass imports/modifications, users with high reputation (many operation earlier in the chain) can add multiple operations in one bundle (w/ one proof-of-work).

Certain useful actions, e.g. blacklisting a known offender, or reclaiming a long-lost key, should be agreed by the community. For that purpose, key owners may vote for motions, the votes are yet another type of operations that are added to the chain. (X% of users, weighted by operations, have to agree for a motion to pass)

Authorized keys for a given identifier can be differentiated by roles. This will enable delegation of concerns to various third-party service providers.
Next steps
Roadmap

- Find interested stakeholders
  (libraries, archives, minting orgs., research centres)
- Secure funding
  (most likely public research funding agencies)
- Implement and deploy
- Maintain and promote
Several parameters in the protocol need to be calibrated. Where applicable, by further research and simulations. In other cases, by consensus.

- Fine-tune the hardness of proofs-of-work
- Establish a catalogue of roles for auth. keys
- Quantify benefits of high reputation
- Agree on the P2P network topology
Thank you for your attention. Let’s stay in touch!

linkedin.com/in/bolikowski

twitter.com/bolikowski

lukasz.bolikowski@icm.edu.pl