Using the DNS as a directory for identities

Vittorio Bertola, FOSDEM 2019
1. The problem
We have too many accounts...

...and we can't even use the same password for all of them.
We already have unified online identities...

...but we do not control them!
Our online identity, today

The big Internet platforms already create an «online identity» for us. They track us across multiple services and sell us for targeted advertising.

Meanwhile, we are stuck with a thousand accounts:
- Insecure, inconvenient etc.
The solution: Single sign-on

SSO = A single set of credentials that can be used on all existing online services

Requires an online service acting as user authentication provider
(must be trusted by everyone)
But of course, the big OTTs already thought of this!
Proprietary SSO gaining ground

Very convenient and ubiquitous
Average Internet users like it a lot

*But*

No interoperability + fragmentation => concentration
Clients have to implement each of them
Users cannot choose their provider
Makes tracking straightforward
Accedi gratuitamente, con uno dei tuoi profili social. Se non ne possiedi uno, scorri la pagina fino in fondo ed usa il pulsante "Registrati"
We need openness and federation!
Advantages of public federated SSO

Why can't your online identity work like your email address?
You only need one account to interoperate with everyone
You get to choose and even change your provider
You can keep your identifier if it is in your own domain name
Advantages of public federated SSO (2)

You only need to remember and secure one set of credentials
Any additional security mechanisms can be implemented just once by a specialized party
You have an easy way to control the sharing of your information and to keep it updated
You don't need to register for new websites, just identify yourself
But *federation* needs a *discovery mechanism*...
What do we miss?

We already have federated identity management and authorization protocols

- OpenID Connect / Oauth 2.0
- Though not normally deployed in a truly federated way (at most, used for a federation with a single identity provider)

We miss a place to keep the directory of all existing identities, and a protocol for looking identities up into it
2.

Where do we keep a public directory for identities?
The Web people do it on the Web

OpenID Connect already has an optional discovery mechanism

- It is based on WebFinger, which is based on HTTPS
- Only accepts URIs as identifiers, with email addresses as a special case

But it requires you to deploy a web server and a WebPKI certificate on each and every domain that you want to use for identifiers
Hey, but the web is so uncool now! Why don't we use a blockchain? Don't you want to be self-sovereign? And by the way, here are some tokens from my ICO!
Transforming digital identity into trusted identity.

Learn how IBM Blockchain Trusted Identity™ is joining forces with others to build the internet’s long missing, decentralized identity layer.

Identity For All
Permanent Digital Identities that Don't Require a Central Authority

Open Identity System for the Decentralized Web
Verified identity decentralized with blockchain technology.
The blockchain people do it on the blockchain

Identities, or at least pointers, or at least hashes, are written into the blockchain

- The rest is often unclear, or proprietary, or vaporware, or all together

The selling point is that this is «decentralized»

- Down with «central authorities»! No government, no ICANN can get in your way!

Unofficial standardization ongoing at the W3C on a «DID» URI scheme
A survey by a potential customer found 91 blockchain ID projects, 63 of which were having an ICO, but only 17 of them had a non-placeholder website, only 3 had downloadable software, and only 0 had working software.

(source: presentation at European Identity Conference 2018)
Wait a minute…

We already have a «public distributed ledger»
It is an open, public standard with many free implementations
It is widely available to everyone everywhere
It has been working reliably for 30+ years
It is secure (if you care to deploy the security extensions)
It can scale effectively to any amount of traffic
It is regulated to prevent capture
It is decentralized and federated
“It’s the DNS!”
The DNS provides the namespace

In the real world, people use «natural» names which are neither unique nor uniform nor easily parsable

So you need a namespace to name identities uniquely on a global scale, while distributing its management... but it's the same problem that was already solved for host names 35 years ago
The DNS provides the namespace (2)

Using the DNS, you can assign human-readable identifiers to identities in a naturally federated namespace.

Users are already familiar with DNS-like strings.

You can even use email addresses if you wish.

Or you can encourage people to get their personal domain name and own a piece of the namespace.
The DNS provides the discovery scheme

We just need a pointer to know who is responsible for an identifier

Again, same problem already solved for email 35 years ago

We use a TXT record, rather than a new RRtype
  □ So we are not adding straw onto the camel’s back

Two Internet drafts independently submitted
_openid.<identifier>

TXT

v=OID1;iss=<issuer>;clp=<claims_provider>
Though, in the end...

This discovery is «blockchain-ready»
- You will just need to replace the DNS-based «public distributed ledger» with a blockchain-based one

However, we want something that can immediately be deployed to mass scale
Or it will be too late to compete with Facebook etc.!
3. The ID4me project
ID4me

Originally promoted by three companies, now a non-profit consortium

All standards are public and patent-free
The recipe for ID4me

1. Take OpenID Connect / OAuth 2.0 – the de facto standard for authorization and authentication over the Internet.

2. Use the DNS to build a distributed, decentralized database of all existing identities and their providers, using “hostnames” as identifiers.

3. Add functionalities on top, extend and publish the standards, create a common ontology for user information, and present to the public.
Roles in ID4me

- **User**: Identity authority
- **Identity agent**: Provides service to user, Manages user relationship, Manages user data
- **Relying party (any online service)**: Keeps and verifies user credentials, Manages consent to data sharing
- **ID4me identifier (any DNS hostname)**: Keeps and verifies user credentials, Manages consent to data sharing
- **Personal information**
- **Credentials and consent**
- **Login confirmation**
ID4me login flow

1. Provide identifier
2. Discover authority and agent
3. Request login
4. Enter password (or be recognized by cookie)
5. Login OK
6. Request user data
7. Send user data
8. Login completed

User

Identity authority

DNS

Relying party (any online service)

Identity agent
Status

Website, public specifications, Java API released
Additional features under development
Started up the international non-profit
A prototype up and running
Beta public launch at Cloudfest (end of March)
Looking for feedback and participation
https://id4me.org/
Thanks!

Any questions?
You can find me at
@vittoriobertola
vb@bertola.eu