Open and federated identities with ID4me

FOSDEM 2020, 2 February 2020

Vittorio Bertola, Open-Xchange
1. The problem
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 SSO

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 can have an easy way to control the sharing of your information and keep it updated

You don't need to register for new websites, just identify yourself
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 (possibly one that does not sell you out)
You can keep your identifier if you buy a piece of the namespace
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?
Why not standard OpenID Connect?

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
Why not blockchain?

We want to be (and we are) blockchain-ready

However, we wanted something that is:

- easily available to any developer and user
- immediately deployable on a mass scale

Otherwise:

- it will be too late to compete with Facebook etc.
- too few people will be able to develop applications and services
"It’s the DNS!"
Why the DNS?

- 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 (with DNSSEC)
- It can scale effectively to any amount of traffic
- It is regulated to prevent capture
- It is decentralized and federated
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
<identifier> = any valid hostname
in a domain that you control

_openid.<identifier>
TXT
v=OID1;iss=<issuer>;clp=<claims_provider>
3. The ID4me project
ID4me

A set of open, patent-free standards

A non-profit consortium for promotion
Roles in ID4me

- **User**
  - ID4me identifier (any DNS hostname)
  - Personal information
  - Credentials and consent

- **Relying party (any online service)**
  - Login confirmation

- **Identity authority**
  - Keeps and verifies user credentials
  - Manages consent to data sharing

- **Identity agent**
  - Provides service to user
  - Manages user relationship
  - Manages user data
**ID4me login flow**

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

**Relying party** (any online service)

**Identity authority**

**DNS**
Status

Website, public specifications, APIs released
Several testbeds up and running
Several authentication plugins available
First ID4me service (Denic ID) being launched
Optional verified identities under development
Started up the international non-profit
  □ 27 members and counting
Coming next

Cloudfest Hackathon project to develop a free «server» (agent + authority) implementation
Standard extensions to provide and manage «strong», verified identities
A public directory for operator reputation
  □ A problem for every federation...
https://id4me.org/

Information, specs, code...
Thanks!

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

Credits: Original presentation template by SlidesCarnival modified by myself
License: This presentation is distributed under a Creative Commons Attribution (CC-BY) license