Making Decentralized Identity Possible with CanDID

IC3

Follow

The Initiative for CryptoCurrencies and Contracts (IC3)

--

Listen

Share

Deepak Maram and Harjasleen Malvai

TL;DR

Decentralized identity systems allow users to gather and manage their own credentials under the banner of self-created decentralized identifiers (DIDs). The key focus of DIDs is on shifting the control of a credential into users' hands. Existing decentralized identity proposals, however, suffer from several problems. First and foremost, how do you bootstrap an ecosystem of credential issuers? It is unlikely that most existing legacy providers suddenly switch and issue such credentials. Second, like with cryptocurrencies, DID systems burden users with managing their own keys creating a significant risk of key loss. They also omit essential functionality, like resistance to Sybil attacks and the ability to detect misbehaving or sanctioned users. We address these problems by introducing CanDID in our

[new paper

](https://eprint.iacr.org/2020/934.pdf).

What is decentralized

identity and why is it important?

Putting existential questions aside, let us consider what it means to have a physical identity or "ID". Consider the following: when obtaining an identity card from a Department of Motor Vehicles (DMV) in the US, a user can plausibly be issued an ID card with the following documents: a social security card (proof of social security status), a birth certificate (proof of date of birth and name), a bank statement, a payslip (proof of name) and a postmarked letter (proof of address). These documents have a name as a common identifier and can therefore be linked to form this person's ID.

Note that health records show date of birth and can often be accessed online. So can bank accounts and payslips. A receipt for an e-shopping package, or a shipping tracker is about as good as a postmarked letter. Finally, the US social security administration provides a portal using your SSN on ssa.gov. This is to say that an identity about as "strong" as what is issued by the DMV could be plausibly issued onlineif

we had the ability to access and verify the authenticity of these sources of data.

Assuming such an identity can be generated online, it can be used in much the same way as a physical ID, such as a driver's license, can be used to avail services. For example, a physical ID card is often needed to create a bank account. Similarly, an online ID could be used to complete "know your customer" (KYC) requirements for creating a cryptocurrency wallet.

Currently, on the internet, a user's identity consists of a unique identifier, along with some statements, or data "stapled" to the identifier. All this data is generated by a single source and controlled by that source. For example, an online healthcare system uses a patient's health insurance card number as her identifier and stores the associated name and demographic data, along with all her test results and medical visits.

These existing systems are problematic in many ways. Firstly, the use of such identity is often limited to the provider that issues it. Even if identity on one site can be proven to others (e.g., using third-party login via OAuth), user privacy is lost in the process since the resulting identity data is heavily monetized in ways users have limited control over. The second serious drawback of server-controlled identity is that it places the responsibility of protecting the private, personally identifiable information of users in the hands of online providers. Naturally, this has led to a long list of devastating breaches. For instance, the data of 143 million customer credit cards leaked in the Equifax breach (2017), the details of 3 billion user accounts leaked in Yahoo breach (2013), etc.

[Backlash against handling of personal data by large tech firms

](https://www.nytimes.com/2021/01/10/technology/tim-berners-lee-privacy-internet.html)has recently given rise to a new method of identity management, known as self-sovereign identity (or) decentralized identity

. Such an identity would not have to be linked to a single centralized data source, which is one aspect of what makes it decentralized.

The key idea is simple, yet powerful: give users complete control of their own data

. In a decentralized identity (DID) system, each user maintains a set of credentials or DIDs. DIDs usually have public and private key pairs associated with them. By controlling the private keys associated with DIDs, users are empowered to disclose or withhold their credentials, as desired, in online interactions. For example, an online job applicant might release a digitally signed credential from her university showing that she has received a bachelor's degree.

Decentralized identity has seen a major surge of interest recently. Several initiatives are working hard to realize the vision of decentralized identity by building a common set of standards to interact with DIDs. The organizations include, but are not limited to W3C DID, Hyperledger Indy/Aries and ID2020.

What is CanDID?

All existing DID proposals however fail to address several key challenges, which we focus in our new paper, CanDID (short for "can

do decentralized identity (DIDs)"). Below we discuss these issues. CanDID leverages a decentralized committee of nodes to issue and manage credentials.

1. Legacy Compatibility

: Most proposed decentralized identity systems presume the existence of a community of issuers of digitally signed credentials. But such issuers may not arise until decentralized identity infrastructure sees use. The result is a bootstrapping problem. A big impediment to DID adoption is the inability of proposed systems to leverage the rich data on users available in existing web services that do not issue signed attestations.

CanDID leverages an oracle — a relayer that provides assurance around the authenticity of data retrieved from authoritative sources, typically web servers accessed via a secure channel such as TLS and allows a prover to prove (publicly or to a particular verifier) that a piece of data originates with a particular source (e.g., as identified by its TLS certificate). CanDID uses an oracle system, either DECO or Town Crier, to allow users to import identities securely from existing systems like social media platforms, online bank accounts, or email accounts. Using this system, the CanDID committee can issue trustworthy credentials without providers needing to explicitly create DID-compatible credentials or even be aware of CanDID. For example, in the paper, we create a SSN (Social Security Number) based credential obtained from the SSA website.

1. Key Management

: To enable users to back up and recover private keys in a secure, user-friendly way, CanDID uses a oracle-based workflow like that for credential issuance. This approach allows users to leverage existing web authentication schemes and engage in a familiar, user-friendly workflow to recover their keys. Users may store their private keys on whatever devices, e.g., mobile phones, they choose for regular use. Users can back up their private keys with the CanDID committee (privately, via secret-sharing) and prespecify recovery accounts on web services of their choice, along with a recovery policy (e.g., successful authentication for 2-out-of-3 accounts). To recover her key, a user proves successful logins under her chosen policy.

1. Sybil-resistance

:

An important requirement for many real-world systems is Sybil-resistance, i.e., how to issue one credential per user? It's fairly straightforward to ensure Sybil-resistance if privacy was not a concern as you could require a user to show her ID, say, SSN to the committee nodes. The problem of course is that it leaks sensitive user ID. So how to achieve Sybil-resistance while protecting user privacy even from the CanDID committee? CanDID uses secure multiparty computation, the technique of computing on secret-shared data, to achieve Sybil-resistance. Please refer to the paper for more details.

1. Accountability

:

It is challenging both to provide user privacy, i.e., conceal users' real-world identities, and achieve compliance with regulations such as Know-Your-Customer (KYC) / Anti-Money-Laundering (AML). Particularly important is an ability to screen users of the system, i.e., identify and bar identified misbehaving or criminal users. Happily, the secure multiparty computation techniques which are useful for Sybil-resistance also allow scanning stored data for newly sanctioned users, in a privacy-preserving way. Additionally, at registration time, a user can show that she is not on a public blacklist (such as a sanctions list) using either a zk-SNARK, trusted hardware, or an oracle.

CanDID opens up a world of possibilities as it can potentially turn any legacy provider into a DID issuer. We are excited to work with J.P. Morgan in building an initial CanDID Proof-of-Concept.

Our full paper, “CanDID: Can-Do Decentralized Identity with Legacy Compatibility, Sybil-Resistance, and Accountability” by Deepak Maram, Harjasleen Malvai, Fan Zhang, Nerla Jean-Louis, Alexander Frolov, Tyler Kell, Tyrone Lobban, Christine Moy, Ari Juels, and Andrew Miller, is available

[here

](https://eprint.iacr.org/2020/934.pdf). Please get in touch with us

if you have an interesting use case for CanDID!

Corresponding author: Deepak Maram (

[sm2686@cornell.edu

](mailto:sm2686@cornell.edu))

We thank Sarah Allen, IC3 Community Manager, for her help in writing this blog post.