Identity Management on the Blockchain

Julian Roos
Advisor: Heiko Niedermayer
Seminar Innovative Internet Technologies and Mobile Communications
Chair of Network Architectures and Services
Departments of Informatics, Technical University of Munich
Email: ga58moy@mytum.de

ABSTRACT
This paper gives an overview over six of the most promising identity management systems that use the blockchain: Sovrin, Jolocom, uPort, ShoCard, Blockstack and Namecoin. It shortly introduces them, briefly explains their design, evaluates them, assesses in how far they fulfill the promises they make and lastly lists their current work and future goals. In the end, all systems are compared with each other regarding some of their properties.

Keywords
Blockchain, Identity Management, Identity Management on the Blockchain.

1. INTRODUCTION
Identity management can be defined as a system to identify, authenticate and authorize individuals or - more generally - identities. It has been an important topic for a long time on and off the internet and with the future of Internet of Things (IoT) devices it seems to become more important. There are a few predictions on how the identity management of the future may look like. The most promising ones rely on the blockchain technology, introduced by Satoshi Nakamoto in 2008 [12]. With the use of the blockchain technology it becomes possible for identity management systems to disprove Zooko's triangle [27]. Zooko's triangle is a theory by Zooko Wilcox-O'Hearn, published in October of 2001, in which he states that three desirable properties for names of participants in a network protocol cannot all be fulfilled at the same time. Those properties are Human meaningfulness, meaning the names are human readable and memorable, security and lastly decentralization.
This is due to the fact that the blockchain technology offers a different approach to storing and managing (digital) identities, mainly because it enables decentralized identity storage without a central authentication authority while preventing tampering with the identities and the data stored. While some proposals use their own blockchain to implement their identity management others rely on already existing ones. Nevertheless, most of those new identity management techniques offer a variety of use-cases and can be used for governments, managing accounts on websites or a new domain name system. For example, they can be used to verify a user's age on multiple websites, without the user having to send each website a photo of his passport.

In this paper we explain, evaluate and compare six identity management systems. The structure of this paper is as follows: Each identity management system is described in its own section and is introduced by a short overview and some of its properties. Then we explain its design in the next subsection. The last subsection begins with an evaluation whether the technology fulfills its promises if it listed some, followed by problems and advantages the technology has and lastly lists their current work as well as their future goals (again only if they are listed). At the end of this paper we then compare the identity management systems with each other over some predefined properties.

2. SOVRIN
Sovrin [21] is a public open source identity network that is built on permissioned distributed ledger technology (DLT). Because it is public everyone can use it and due to it being a permissioned DLT nodes that form the consensus have to be authorized by the Sovrin Foundation. Sovrin enables its users to have a self-sovereign identity [22], which means that the users have lifelong full ownership of their identity and do not rely on any central authority to store it Additionally, the identity is private which means they can manage it themselves and can choose to whom they reveal what information (and to whom not) [3] [1].

2.1 Design
A Sovrin identity uses decentralized identifiers (DID) [6] to enable the identity and binds them to a user by using asymmetric cryptography. The DIDs require no central authority to be issued and enable users to create identifiers that are permanent, globally unique and cryptographically verified while the identity owner maintains full control over those identifiers. Those DIDs are then put onto a blockchain together with a DID document object (DDO) comprising the identity owner's respective public key (for this DID), other information the identity owner wants to reveal as well as the network addresses needed for interaction. The identity owner owns the DDO by having the respective private key. Therefore, everyone with access to the internet can verify his control of the private key and consequently of the DID. Additionally, there is no limit of the number of DIDs a user can have, so one can create as many as needed to ensure privacy.

2.2 Evaluation
On their website the Sovrin Foundation claims that "Sovrin identities will lower transaction costs, protect people's personal information, limit opportunity for cybercrime, and

doi: 10.2313/NET-2018-11-1_14

simplify identity challenges in fields from healthcare to banking to IoT to voter fraud" [22].

The argument for lower transaction cost is that since the seller can have more information about a customer he can reduce his risk and therefore his price. An example of this could be a car renting firm, where the customer can show his driving record i.e. the record that shows all received punishments that are driving related if he has it attached to his identity and wants to show it. The car renting firm's employee could now check whether the customer has any major driving violations or if (supposedly) is a safe driver. If the customer appears to be a safe driver, the risk for the car renting firm reduces and therefore it can offer a better price. This only works if the verifier trusts the authenticity of the information, otherwise the price will stay the same. The protection of people's personal information is also given, since people themselves can choose what to share. If someone does not want to share anything, one is not forced to do so and therefore one's privacy is not infringed. However, it is possible that for example sellers will not sell anything if a user does not expose her name. In this case the user is not entirely forced to reveal it, but in order for her to get the item she needs to do that.

The claim that it limits the opportunities for cybercrime can be linked to the fact that Sovrin improves the identity and access management. This results in less accounts being stolen and used for cybercrime. Furthermore, Sovrin enables a seller to only sell to buyers that have shown him their identity and whose identities are validated by someone the seller can trust. In that case, the chance of a buyer committing fraud is mitigated since he can be identified easily. Therefore, also this claim is valid.

Lastly, the Sovrin foundation say that Sovrin will simplify identity challenges in various areas. This mainly depends on the amount of people that use Sovrin identities. If it surpasses a critical mass it becomes suitable in those areas to include Sovrin identities to face their challenges. If it only has a relatively low number of users it may not be useful to include it. Nonetheless, one can say that the possibility to store identity attributes that are hidden until their owner chooses to reveal them offer some great possibilities in those areas.

In conclusion, Sovrin does not promise false claims and most of them can be backed, however, the last one seems to be a bit exaggerated considering that it predominantly depends on the number of users Sovrin has.

3. JOLOCOM

Jolocom [10] is an open source project that is currently developing an identity management system that uses hierarchical deterministic keys (HD keys) to offer decentralized identity to its users. Jolocom identities are also self-sovereign.

3.1 Design

The HD keys are generated from a known seed and controlled by the users [10] and enable the user to generate further child keys from the parent key [17], which can be recovered later if the seed is known. This enables users to generate multiple "personas" which are basically sub identities that can offer anonymity in some interactions. In addition, the child keys make it possible to have ownership of IoT devices mapped onto the Jolocom identity. However, an implementation of an identity management system that just uses HD keys has usability issues [10], therefore Jolocom combines HD keys with DIDs. A DID is derived from the user's public key and the corresponding DDO is stored on an InterPlanetary File System (IPFS) which is a decentralized distributed file storing system. Then the mapping of the DID to the hash of the DDO's IPFS will be stored in a smart contract on the Ethereum blockchain.

To store identity attributes Jolocom uses so-called "verified credentials" that are files that contain a statement and signature by the verifier identity. An example used in the Jolocom whitepaper for verified credentials is: "This user is employed by Company X" as a statement and a signature containing metadata about that statement by the verifier identity - in this case Company X.
For storing statements, the authors distinguish between private and public verified credentials as well as the need for constant availability. Private verified credentials usually require access control and are stored on self-hosted servers when they should be constantly available and on the user's device when they do not have to be constantly available. Public verified credentials with constant availability should be stored on distributed storage, for example IPFS. In the case of non-constant availability, the authors list no storage proposal because there is "no apparent use case for this configuration" [10].

3.2 Evaluation

Besides the claim for self-sovereign identity Jolocom does not promise any additional benefits of its technology - neither in the paper nor on its website. Since Jolocom already has a working application it therefore fulfills its claims.
However, there is one thing that Jolocom does not do (and does not claim to do) that is not entirely clear: In their conclusion the authors write: "We offer a truly self-sovereign decentralised digital identity solution" [10]. This could make someone believe that Jolocom eliminates all need for trusted third parties (TTPs). But this is not the case since Jolocom does not remove the need for a TTP when it comes to authenticating the attributes and getting verified credentials. An attribute itself is worthless unless it got verified by someone who is trusted - so a TTP. The easiest example is that of age: Anyone could make a Jolocom identity and claim to be 18 years or older to be able to buy things restricted by age. The claim itself is worthless, unless a government agency, bank or other TTP signs it, thus making it a verifiable credential.

Their future goals comprise providing a simple, global and self-sovereign identity for everyone while continuing to use the existing technical standards regarding decentralized digital identities and keeping open source releases. Furthermore, the implementation of the storing of private verifiable credentials that should be constantly available is not a priority right now, and also a future goal.
Currently Jolocom is focusing on the storing on private verified credentials that are saved on the user's phone and the public verified credentials.

doi: 10.2313/NET-2018-11-1_14

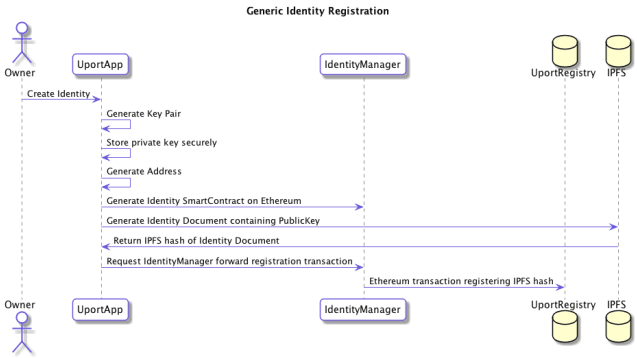

Figure 1: Identity creation process. Retrieved from [24]

4. UPORT

uPort is an open source identity management system that is used on the decentralized internet [26]. It claims to enable users to have a self-sovereign identity registered on the Ethereum blockchain which can then be used for some applications like the usage of credentials or managing of data. Currently, uPort exists only as a mobile application [25].

4.1 Design

To implement its self-sovereign identities uPort uses smart contracts on the Ethereum blockchain. Smart contracts are stored on the Ethereum blockchain and are a piece of code that can move ether (Ethereum's cryptocurrency) as well as data when invoked [8] [7]. To be executed they have to be called by their unique 160-bit hexadecimal identifier [7]. uPort uses two smart contracts on the Ethereum blockchain that are called controller and proxy.
The whole identity creation process is depicted in figure 1. To create an identity, an asymmetric key pair is created and then an instantiation of the controller with a link to the just created public key is made. After that a new proxy is created with a link to that controller instance and only that controller instance is able to invoke the proxy's functions. This can be seen in figure 2.The uPort identifier (uPortID) is now the address of the newly created proxy [7]. There is no limit to the number of uPortIDs a user can have except the number of possible 160-bit identifiers of the proxy smart contract, which obviously is 2^{160}. Furthermore, all created uPortIDs for a user are unlinkable if one does not link them oneself. To store identity attributes a global smart contract called registry is used. The registry stores a hash of the JSON attribute structure together with a uPortID. To access the attributes, which are stored outside of the blockchain on an IPFS, the stored hash is used.

Because the smart contract is global and every hash for every uPortID is stored there everyone can access the attributes of a uPortID. However, only the owner of the uPortID can change its own stored attributes. This system enables uPort to also support verified credentials.

Because the private key is only stored on the user's mobile phone which can get lost, uPort has a system that enables its users to regain control of their identity with a different key. The system works as follows: The user nominates a group of

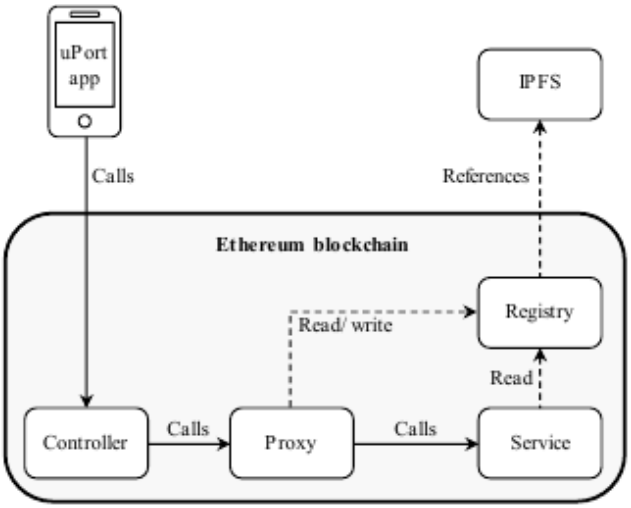

Figure 2: Interactions between smart contracts. Retrieved from [7]

```
{

   "@context": "http://schema.org",
   "@type": "Person",
   "publicKey": "0x044c31ed1499dce76ee7711c7238...",
   "publicEncKey": "Py+NXzHgacNMTzj9Ufe4S2KPuzR...",
   "name": "First Last"

 }
```

Figure 3: Example of a uPortID stored as a JSON attribute structure. Retrieved from [24]

trustees (with their uPortID) in advance. The trustees can then vote to change the associated public key of the user's uPortID with a new one and the key will be changed once a quorum is reached. Note, that there is no way for the system to identify whether the private key was truly lost. Therefore, it is possible for malicious trustees to take over a uPortID if they get quorum.

4.2 Evaluation

The system uPort offers is working since it is based on the already working Ethereum smart contracts. One is able to create an identity, manage it together with its respective attributes and can get control back in the case of loss due to the trustee system. However, the trustee system can be exploited if the uPortID has too many malicious trustees. Nevertheless, the system is working since there is no other way to regain control of a uPortID and it is the user's fault when he nominates too many malicious trustees. If uPort had not introduced the possibility to regain control of a lost uPortID, then it would not be a good identity management system and possibly be flooded by "dead" accounts which no one has control over anymore. Although, this is still possible, if a user loses his private key and has not named any trustees before.
Additionally, like Jolocom, uPort also uses verified credentials, therefore, a user is still depended on TTP. Because

doi: 10.2313/NET-2018-11-1_14

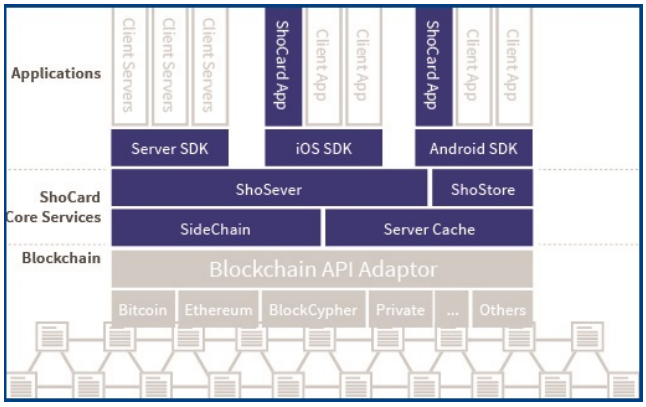

Figure 4: ShoCard's architecture. Retrieved from [19]

uPort has no promises other than a working uPort system, uPort meets all its requirements.
Currently uPort is working on the development of a mobile SDK, so that uPort accounts can be implemented in every app.

5. SHOCARD

ShoCard [20] [19] combines identity management on the blockchain with already trusted credentials like passports or driver licenses. The identities can then be extended with further attributes. Furthermore, is ShoCard the only identity management system in this paper that uses its own server to store relevant information for the identification of users (although ShoCard has no access to those because they are encrypted). An advantage of ShoCard is, that it is able to use multiple blockchains at the same time and can add new ones if needed [19].

5.1 Design

ShoCard's architecture can be seen in figure 4. To create a ShoCard identity or ShoCardID a user needs the ShoCard mobile application. The app generates an asymmetric key pair and then the user has to take a picture of the user's passport as well as a picture of himself [18]. Afterwards the selfie and the passport data are encrypted together and then stored locally on the user's device. Furthermore, the passport data and selfie are each hashed, signed (with the just generated private key) and then put together into a seal record. The seal record is then put onto the blockchain and can then be retrieved through its unique address on the blockchain. This unique address is at the same time the user's ShoCardID.

Now comes the certification part in which the information of the user gets certified. In the context of paper [18] this is done in the context of air traveling by an airline agent or automated kiosk. The user now has to show that he has control over the seal stored on the blockchain as well as the key that signed it. For this the user has to show the original data that were used to create the seal and sign a challenge to show ownership of the private key [7] [18]. Moreover, the user has to present his identification (passport, driver license, etc.) to an agent who will check that the official information of the identification matches the ones of the ShoCardID. The agent will further check if the selfie shows the user. If everything is right, the agent will request a certification of the user that certifies the ShoCardID and selfie with the airline's private key. Note that if the checking is done by an automated kiosk and without a human agent, it will look a bit different. Nevertheless, the same things will be checked and the user will also be certified at the end. Afterwards, a message containing a reference to the ShoCardID, the certification records and some further information is created, signed with the user's private key, encrypted with a new symmetric key and then stored on the ShoStore servers. This is called enveloping and the stored data can be accessed with an EnvelopeID. During this process ShoCard never learns the symmetric key and therefore only the user is able to share the data.

If the user now needs to be verified, he can generate a QR-code containing the EnvelopeID as well as the symmetric key. The agent now scans the QR-code and the envelope is downloaded from the ShoStore server and then decrypted when the download is finished. The now following verification consists of i) comparing the private key used for signature on the envelope as well as the seal record, ii) comparing the passport information and selfie versus the seal record and lastly iii) validating the signing of the certification records [18].

5.2 Evaluation

On its websites ShoCard states in the about ShoCard section "It's the one identity verification system that works the way consumers and businesses need it to for security, privacy, and always-on fraud protection" [20]. Even though the ShoCard system is working, it relies on many trusted authorities to be enabled. For example, in the use case of air travel one needs to trust every airline since every airline can certify information for any user. This can become a big problem when there is no knowledge about the authenticity of an airline or if it actually exists. Because in the latter case attackers could fabricate an airline and then sign their own information while they could bribe smaller but real airlines in the former one. This problem can be avoided if one does not trust any other airlines or diminished when airlines just trust big airlines. However, the first case would make the technology completely useless while the latter one would shrink its use by a bit, depending on the number of passengers that fly with one of the non-trusted airlines.

Another aspect is the storage of the envelope on the ShoStore servers. If the servers are down for whatever reason, the validation of a ShoCardID cannot be done which basically renders ShoCardIDs useless during that time. This becomes extremely problematic in the case of a permanent shutdown of the servers (e.g. if the company stops existing) because then ShoCardIDs are straight up useless.

A huge advantage of ShoCard is the fact that it can use multiple blockchains and change to new ones if it needs to. This is due to most other systems relying on one (specific) blockchain and would cease to exist if that blockchain would stop existing as well or if it somehow got rendered useless by a new attack.

doi: 10.2313/NET-2018-11-1_14

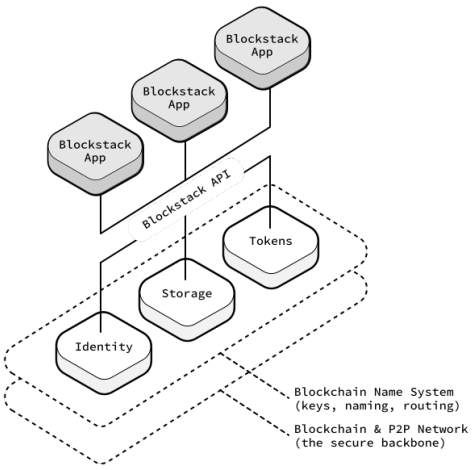

Figure 5: Blockstack's architecture. Retrieved from [5]

Concluding, one can say that while ShoCard is working, there are other identity management system out there that can also verify a user's information. The problem whether one can trust the verifying instance remains the same in all those identity management systems, including ShoCard. With the central server needed for ShoCard and the resulting availability problems it does not look all too promising for ShoCard. On the other hand one has to admit that ShoCard is focusing on the consumer business interaction while the other systems have a more general approach for identity storage. In the end it is up to the businesses if they want to risk having an unusable system in the case of ShoStore being unavailable for the advantage that ShoCard is specifically being developed for that area as well as being independent from one specific blockchain.

6. BLOCKSTACK

The main goal of Blockstack is a decentralized internet where users do not need to trust remote servers and can run decentralized applications which can be seen in figure 5. To implement this Blockstack claims to have a solution to replace some of today's core internet infrastructure like DNS, public key infrastructure and storage backends. Blockstack is open source, can work on any blockchain and in addition, also offers identity management on its blockchain called blockstack identities [4].

6.1 Design

The blockstack identities consist of profile and globally unique names and can be registered for people, companies, websites, software packages and more [4]. Each profile can contain private as well as public information, both of which are put in by the identity owner. Other peers and select authorities can then validate this information.

Blockstack also implements a direct use for decentralized identities based on the blockchain with Gaia, its decentralized storage system [2]. The storage of data on Gaia has comparable speed to today's cloud services. That is the case because Gaia uses the infrastructure of current cloud providers (like Dropbox, Amazon S3 and Google Drive) to store encrypted and/or signed data on them. The stored data are tamper resistant because the data hashes are stored on the blockchain. On Gaia a user can use a decentralized identity based on the blockchain to log in to apps and services, which enables him to save data generated by those on storage backends under the user's control.

In order to replace DNS, Blockstack proposes a new system based on the Blockchain called Blockchain Name System (BNS). BNS provides similar functionality to DNS but does not have any central root servers. The naming system in BNS relies on names being owned by cryptographic addresses of the blockchain and their respective private keys. In order to be able to register a website, one has to first preorder it. This is done to prevent a race attack, in which the attacker tries to register the same name while the transaction is still unconfirmed. A website belongs to the user who was the first one to successfully write both, a preorder and register transaction. The names in the BNS are structured into namespaces whose information are stored on the root blockchain. Namespaces define the costs for names as well as their renewal rates and are therefore similar to the top-level domains (TLDs) in DNS. The resolvement process is as follows: The user that wants to resolve a name, makes a query to the BNS server that runs in her trust zone. Then the BNS server looks at the relevant blockchain, searches the relevant record and retrieves the respective zone file from the linked but untrusted external source.

6.2 Evaluation

The authors state a variety of different desirable properties in the paper [2]: The first thing is that "End-users should be able to (a) register and use human-readable names and (b) discover network resources mapped to human-readable names without trusting any remote parties". The first aspect gets enabled through the blockstack identity while the other aspect is being made possible by the BNS system.
Further they state that "End-users should be able to use decentralized storage systems where they can store their data without revealing it to any remote parties". This gets enabled through Gaia, where encrypted data gets stored in multiple clouds.
Lastly they want that "The end-to-end performance of the new architecture (including name/resource lookups, storage access, etc.) should be comparable to the traditional internet with centralized services". As already stated, this is true for Gaia, the decentralized storing system. There is no time comparison included with the BNS, though it needs less requests when resolving a name. However, it is still possible that it takes a longer time, even with less requests.

The first property (a), that names should be human-readable, is also the reason why Blockstack (and Namecoin) solve Zooko's triangle in contrast to Sovrin, Jolocom, uPort and ShoCard. This is due to the fact that in the latter ones the names are generally not human-readable. So, while all identity management systems in this paper are secure and decentralized (the two other properties of Zooko's triangle), only Blockstack and Namecoin also enable human-readable names. For example, in uPort the uPortID is the address of its corresponding proxy smart contract on Ethereum's blockchain which is a 160-bit number that is typically repre-

doi: 10.2313/NET-2018-11-1_14

sented as a 40-character hexadecimal string. In contrast, in Blockstack a name can be chosen by the user and the only requirement is its uniqueness.

The fact that Blockstack is able to use any blockchain is very important for further success. Because due to it, Blockstack is able to migrate from one blockchain to another, in case a blockchain becomes insecure in any way e.g. if not enough hash rate is in the system which enables malicious actors to do a 51% attack. However, it is not as secure as ShoCard's approach that supports multiple blockchains at the same time.
Another big advantage is that it already has a lot of registered identities. Currently Blockstack has 80,000 registered identities [4] and therefore is one of the biggest systems with identity management on the blockchain.

7. NAMECOIN

Namecoin is an open source technology that aims at improving decentralization (and therefore resistance of censorship), security and privacy [14]. Additionally, it also tries to improve certain structural aspects of the internet like the domain name system (DNS) and identities. The use cases listed on their website include censorship resistance, attaching attributes to an identity, decentralized TLS certificate validation based on the blockchain and the use of the .bit top-level domain (TLD) [14]. Namecoin is the first fork of the cryptocurrency Bitcoin, which was the first application of a blockchain, and consists for a great part of the Bitcoin codebase with some added functionality.

7.1 Design

Namecoin enables the user to register names and then store associated JSON values (up to 520 bytes) on their blockchain. The Namecoin software can then be used to search the blockchain for the name and retrieve the respective data (values). An identity can consist of a name, email address, photo (in the form of an URL due to the limited storing space), crypto address (like Bitcoin and Namecoin address), fingerprints of cryptographic keys and other things [15]. Unlike most other identity management systems, identities (or rather the names under which they are stored) have to be renewed at least every 35,999 blocks which is approximately every 200 - 250 days [13].

A Namecoin identity has a name and normally a connected email address and Namecoin address. Additionally, there is also a fingerprint of some cryptographic key stored. This data can then easily be fetched (if one knows the id) which can be seen in figure 6. Furthermore, Namecoin is the basis of NameID, a service that enables one to use his Namecoin identity to create an OpenID which can then be used to log into OpenID enabled websites.

7.2 Evaluation

Namecoin is already working and is able to fulfill most of its promises. Especially in the range of identity management, Namecoin does what it is supposed to do. Like Blockstack, Namecoin also solves Zooko's triangle, because a user can choose his name himself. Although, its attribute storage space for identities is only 520 bytes at maximum, it enables the use of URLs that can then contain almost infinite storage

```
$ namecoind name_show "id/khal"
{
  "email":    "khal@dot-bit.org",
  "bitcoin":  "1J3EKMfboca3SESWGrQKESsG1MA9yK6vN4",
  "namecoin": "N2pGWAh65TWpWmEFrFssRQkQubbczJSKi9"
}
```

Figure 6: Fetching data from the Namecoin identity khal. Example retrieved from [15]

space. But the information stored on the site of the URL still need to be present (e.g. as a hash) in the Namecoin identity, to disable tampering of them.
However, Namecoin is not widely used, hence, the .bit TLD is not really common. Even though Namecoin wants to decentralize the DNS system, their next goal is to get most nameservers to implement the .bit domain because otherwise there is no real growth for websites in the .bit domain. This is due to the fact that one currently needs to install Namecoin and then download the whole blockchain to visit those sites. Another possible solution that is currently explored and does not depend on the nameservers to implement the .bit TLD is the inclusion in a browser or OS. Furthermore, the registration of the .bit TLD as a special use domain, like TOR's .onion TLD, is tried.

8. COMPARISON

In this section we will - if possible - compare the different systems with each other by looking at their properties. Those properties are:
1. Usage of a permissioned blockchain?
2. Do they offer self-sovereign identities?
3. Built-in incentives for the nodes that run the blockchain to stay honest?
4. Who can use the identities and who can verify them?
5. Do they offer more than identity management?
6. Their current development status

The first property (1) is the use of a permissioned blockchain i.e. a blockchain for which nodes need permission to work on. A system that has a permissioned blockchain is Sovrin, where the nodes need to be certified to work on that blockchain. Jolocom and uPort use the Ethereum blockchain, which is not permissioned. ShoCard and Blockstack are not bound to one specific blockchain and therefore do not need to run on permissioned blockchains. Namecoin is running on its own fork of the Bitcoin blockchain and because the Bitcoin blockchain is not permissioned neither is the one of Namecoin.

The next property (2) is the one of self-sovereign identities. Sovrin, Jolocom and uPort claim to offer self-sovereign identity, and Sovrin as well as Jolocom fulfill all criteria for it. But uPort has a problem with the user's attributes, because anyone with the uPortID can see them. Because one has no choice over whom to reveal what attributes to, it does not satisfy all our requirements for self-sovereign identity. In contrast, ShoCard, Blockstack and Namecoin do not claim to offer self-sovereign identity. ShoCard does not enable its users to have a self-sovereign identity be-

cause it has a central authority (the ShoStore servers) used to store data which contradicts the conditions for a self-sovereign identity. Blockstack seems to meet all criteria needed for self-sovereign identity. However, since identity management is not the focus of Blockstack its implementation is not described. Therefore, it could be the case that it does not satisfy some criteria when it comes to privacy. Namecoin's problem for self-sovereign identity is the same as uPort's. The identity is stored as unencrypted JSON values on Namecoin's blockchain and thus anyone can access the data. Therefore, Namecoin does also not fulfill this criteria for self-sovereign identity.

Aspect (3) is about built-in incentives for the involved actors (mostly nodes running the blockchain) to stay honest which is important for a flawlessly working long term system. Sovrin does not offer incentives to help sustain the network, however, due to the nodes running the blockchain being trusted the risk of nodes being malicious shrink significantly. Both Jolocom and uPort use the Ethereum blockchain which offers mining rewards in the form of ether to its nodes [9]. ShoCard and Blockstack can work on different blockchains and therefore have the possibility to switch to one with rewards for miners. In Namecoin the miners also get rewards in its cryptocurrency NMC [13].

The following aspect (4) considers who can use the identities and who can verify them. In Sovrin anyone can create identities and anyone with access to the internet can verify them. Jolocom enables anyone to create an identity and any claim by a Jolocom identity can be verified using the Jolocom's user interface. With uPort everyone can create an identity and everyone can verify the claims that identity made by verifying the signature of that claim. ShoCard also makes it possible for everyone to create an identity, but the verification is done through some verifiers depending on the use case. In its most prominent example the verifiers are airline agents or automated machines of the airline. In Blockstack everyone can have an identity, but it may happen that the respective name is already taken and one has to take a different one. Blockstack claims that other peers and select authorities can then verify this information, but do not describe a specific process. Namecoin, like Blockstack, also enables anyone to have an identity but the names of the Namecoin identities are unique. Namecoin does not directly offer a method to verify a name identity. However, if the other person included the fingerprints of some cryptographic keys one could check if the other person is in control of those keys.

In order to classify the identity management system, aspect (5) is about whether the system offers more than just identity management. The only systems that provide more than identity management are Blockstack and Namecoin with their approach to a decentralized internet / DNS system. The other systems provide only identity management.

The last aspect (6) is the current development status of the systems i.e. what they currently offer. This is important to assess the likelihood of a system having widespread support. It is notable that most systems are developed as open source projects. For Sovrin we found no possibility to use it yet and create an identity at the current time, so it still seems to be in early development. Jolocom released the alpha version of its SmartWallet in the Google Play Store at the end of February [11]. With the app users are able to create a decentralized identity with verified credentials, just like Jolocom proposes. uPort already launched its uPort ID mobile app alpha in January 2017 and is working with the city Zug (Switzerland) to officially register their citizens [23]. The current working area is the mobile SDK to enable other apps to use uPortIDs and get a widespread support for them. ShoCard currently offers a demo version on its website, that one has to request. Furthermore, ShoCard Inc. offers two apps in the Google Play Store, ShoCard and ShoBadge. ShoBadge is a digital identity card for mobile application. Blockstack is already working and has (by its own claim) 80.000 registered identities. Similarly, Namecoin is also working and has implementations like NameID [16] that enables users to turn their Namecoin identities into an OpenID.

9. CONCLUSION

We presented six identity management systems in this paper. They vary in terms of different design and sometimes a different target group. Sovrin has its permissioned blockchain, Jolocom its HD-Keys to offer sub-identities, uPort utilizes smart contracts and ShoCard does not rely on one Blockchain and is further focusing on the consumer business interaction. Blockstack and Namecoin differ from the rest because they want to decentralize the internet and from each other due to Blockstack's wider approach in decentralizing the internet with decentralized DNS, storage and applications while Namecoin only proposes a decentralized DNS.
There are quite a few advantages that the identity management systems presented in this paper have over the current ones. Most notably those are: more control over one's identity and in some systems self-sovereign identity, a decentralized identity due to blockchain and easier verification to multiple entities. However, the systems still rely on TTP's to verify identity attribute claims, otherwise those claims are mostly useless.
Concluding, one can say that all described identity management systems have their special and unique properties and it remains to be seen whether identity management on the blockchain will be the identity management of the future and if any of the systems presented in this paper will succeed.

10. REFERENCES

[1] A. Abraham. Whitepaper about the concept of self-sovereign identity including its potential. October 2017. https://www.egiz.gv.at/files/download/Self-Sovereign-Identity-Whitepaper.pdf.

[2] M. Ali, R. Shea, J. Nelson, and M. J. Freedman. Blockstack: A new internet for decentralized applications. October 2017. https://blockstack.org/whitepaper.pdf.

[3] C. Allen. The path to self-sovereign identity, April 2008. http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html.

[4] Blockstack - Blockchian Identity. https://blockstack.org/posts/blockchain-identity.

[5] Blockusign. https://blockusign.co/signup.html.

[6] Decentralized Identifiers (DIDs) v0.10. https://w3c-ccg.github.io/did-spec/.

doi: 10.2313/NET-2018-11-1_14

[7] P. Dunphy and F. A. P. Petitcolas. A first look at identity management schemes on the blockchain. https://arxiv.org/ftp/arxiv/papers/1801/1801.03294.pdf.
[8] Ethereum-Whitepaper. A next-generation smart contract and decentralized application platform. https://github.com/ethereum/wiki/wiki/White-Paper.
[9] Ethereum Wiki, Mining Rewards . https://github.com/ethereum/wiki/wiki/Mining#mining-rewards.
[10] C. Fei, J. Lohkamp, E. Rusu, K. Szawan, K. Wagner, and N. Wittenberg. Jolocom: Decentralization by design. February 2018. https://jolocom.com/whitepaper/.
[11] Jolocom - 2018 Where we have been so far. https://jolocom.com/2018-where-weve-been-so-far/.
[12] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008. https://bitcoin.org/bitcoin.pdf.
[13] Namecoin FAQ. https://namecoin.org/docs/faq/.
[14] Namecoin Website. https://namecoin.org.
[15] Namecoin Wiki. https://wiki.namecoin.org/index.php?title=Welcome.
[16] NameID Website . https://nameid.org/.
[17] K. Robles. Hierarchical deterministic keys. https://www.w3.org/2016/04/blockchain-workshop/interest/robles.html.
[18] ShoCard. Travel identity of the future. 2016. https://shocard.com/wp-content/uploads/2016/11/travel-identity-of-the-future.pdf.
[19] ShoCard. Shocard whitepaper - identity management verified using the blockchain. 2017. https://shocard.com/wp-content/uploads/2018/01/ShoCard-Whitepaper-Dec13-2.pdf.
[20] ShoCard Website. https://shocard.com/.
[21] Sovrin-Foundation. A protocol and token for self-sovereign identity and decentralized trust. January 2018. https://sovrin.org/wp-content/uploads/Sovrin-Protocol-and-Token-White-Paper.pdf.
[22] Sovrin Website. https://sovrin.org/.
[23] uPort - 2017 Recap, 2018 Outlook . https://medium.com/uport/uport-year-in-review-whats-to-come-in-2018-15ccb9214439.
[24] uPort Developer Website. https://developer.uport.me/.
[25] uPort Specs. https://github.com/uport-project/specs/.
[26] uPort Website. https://www.uport.me/.
[27] Z. Wilcox-O'Hearn. Names: Decentralized, secure, human-meaningful: Choose two. October 2001.

doi: 10.2313/NET-2018-11-1_14