Identity Management and KYC using Blockchain Technology

by Morten Helles, CTO co-founder, NewBanking

International Blockchain Conference, Sydney, April 6th 2017
Table of Contents

1. Me

2. NewBanking

3. Key Questions and Considerations on Blockchain Usage
   a. Why use blockchain?
   b. Who to have access to the blockchain?
   c. What to store (and not to store) on the blockchain?
   d. How to manage security?
   e. Which blockchain technology to use?

4. Conclusions
Who am I?

Morten Helles, Danish and reside in Copenhagen, Denmark
M. Sc. in Computer Science, Bach. in International Business
IT Entrepreneur and Board Advisor
CTO and co-founder of NewBanking
Member of Danish Standards and part of ISO TC 307 on Blockchain
Who is NewBanking?

NewBanking is a Danish startup company that offers a solution to two problems related to digital identities.

1. The enterprise AML/KYC problem.
   a. It is difficult and expensive to do properly. And its regulated.
   b. Privacy legislation and the "Right to be Forgotten."

2. The customer's complexity problem.
   a. Going through many different KYC processes.
   b. Personal data is spread out. Not clear who holds what personal data.

The solution: A customisable "KYC-as-a-service" for enterprises, and a digital identity and data access management platform for private users.
NewBanking's Value Proposition

Customer Value Proposition

- Easier and cheaper to be(com) compliant with AML/KYC regulation.
- Each customer configures his own unique KYC process: methods and flows.
- KYC data is collected efficiently and stored securely, hosted by NewBanking.

Private User Value Proposition

- Enter KYC data only once, and manage it in one place. (Big time saver.)
- Control who can access KYC data. ("It's my data, and I choose who to share it with.")
- Proof of access grants and revokes.

Important! KYC is a regulated process and so regulators will need access to certain users' data in case of e.g. a KYC audit or an AML investigation. Of course, such regulators must only access data of users whose activities fall under their jurisdiction.
NewBanking's Main Data Types

KYC data, such as name, email, address, phone, passport, social id, etc.

Highly confidential data that must be kept safe at all times.

Verification data for each piece (or set of) of KYC data

Used to document the KYC data verification process (for e.g. auditing purposes).

Grant data that gives customers access to private user's KYC data.

Used to control who can see what KYC data.

"Secret data" including encryption keys and public-private key pairs.

Used to keep data safe and confidential, and for data signing.
Key Question 1: Why Blockchain?

We needed a secure and trusted way of collaborating on digital identities, not only with enterprises and private users, but also with authorities and regulators.

Blockchain characteristics of most interest to us:

- A Single Source of Truth - we cannot disagree
- Security - it's practically impossible to "cheat"
- Provenance - who did what when?
- Transparency - everybody (with proper access) can see what's going on.
Key Question 2: Who to Access the Blockchain?

Who should have and run a blockchain node?

NewBanking and customers.

It is not feasible for private users to run a blockchain node.

How should consensus be reached?

By having all parties participate in "mining".

By using a non-competitive consensus mechanism (i.e. not proof-of-work).

Who should be able to read and write?

Only NewBanking can write data to the blockchain. (To start simple.)

Customers can only access data that is relevant to them.

Authorities can - upon request - get access to certain user data.
Key Question 3: What to Store on the Blockchain?
Should we store KYC and verification data on the blockchain?

Benefit: Customers can access it directly from their local blockchain.

Using the "share data design pattern" using sub-chains (or key-value streams):

Stream 1 "pubkey" contains the public key of all accessors.

Stream 2 "data" contains data encrypted with secret.

Stream 3 "grant" contains secret encrypted by accessor's public key.

Problem 1: Blockchain bloat.

Problem 2: What if secrets (encryption keys) are compromised?

Problem 3: What if user wants to revoke a grant?

Our conclusion: Don't store confidential data. Instead, customers can retrieve confidential data via a secure API.
Key Question 3: What to Store on the Blockchain? (cont.)

Should we store Grant data on the blockchain?

Benefit: Proof (and notification) of who was granted what data access and when.

Problem: How to avoid other customers seeing other grants than their own? (i.e. to avoid or minimize any confidentiality leak.)

Solution

Using customer-specific streams with access control.

Have no confidential data in the grant itself: No user ids, no KYC data, but only a grant id and KYC content hashes (to act as proof against tampering).

Encrypt the grant with a customer-specific encryption key.
Key Question 4: How to Manage Security?

"The Bitcoin Approach": Users can manage their own (private) keys, and they can interact directly with the blockchain.

Main benefit: Security, with minimum impact in case of a security breach.

But with KYC, regulators must be able to - under special circumstances - to access a user's data.

Also, most end-users cannot manage their keys safely. (In Bitcoin, ~30% of all bitcoins are lost.)

Our Approach

Only NewBanking can write to the blockchain. NewBanking controls blockchain permissions.

End-users interact with the blockchain via NewBanking.

Customers can only read "their own" data on the blockchain.
Key Question 5: Which Blockchain Technology to Use?

Our requirements

- Easy-to-use (because blockchain is complicated enough as it is)
- Permission management (to control who can do what)
- Sub-chains or streams (for "grant messaging")
- Clear API and documentation
- An effective consensus mechanism (not based on resources)
- Open source (for trust)
- Native asset support. (Since we will be adding payment monitoring services later.)

We ended up choosing MultiChain, a permissioned Bitcoin-fork for the enterprise.
Conclusions

Blockchain is a complicated technology to work with, e.g. in terms of security architecture, nothing can be deleted, and little in terms of best-practices.

"Think Big, but Start Small" to get some hands-on experience.

Mixing blockchains and traditional (centralized) databases might be a natural and good way to leverage the different strengths of each.

In our case, the blockchain is used for grant provenance, notification, transparency, and single source of trust. A centralized database holds the confidential (and encrypted) user data.

Limitations due to lack of features, e.g. fine-grained permission control to delete (or otherwise make inaccessible) compromised or erroneous data.

In the longer-term: Need for standards on e.g. blockchain interoperability that will allow our KYC'ed user identities to be used on other blockchains.
Thank You