Securely Migrate Digital Identities from a Class PKI to a Blockchain

Keywords: Certificate authority, Digital identity management, PKI, Blockchain

Reading list:

Bitcoin: A Peer-to-Peer Electronic Cash System
https://bitcoin.org/bitcoin.pdf

Greg Slepak on HTTPS, Identity and DNSChain:
https://www.youtube.com/watch?v=W4faDEyHJeM

Blockstack: A Global Naming and Storage System Secured by Blockchains
https://www.usenix.org/node/196209

Problems with the current class PKI

- Single point of failure

  One certificate authority can undermine the security of the whole system.

- Poor identity retention

  A single user can have multiple public keys.

- Expensive as f*ck

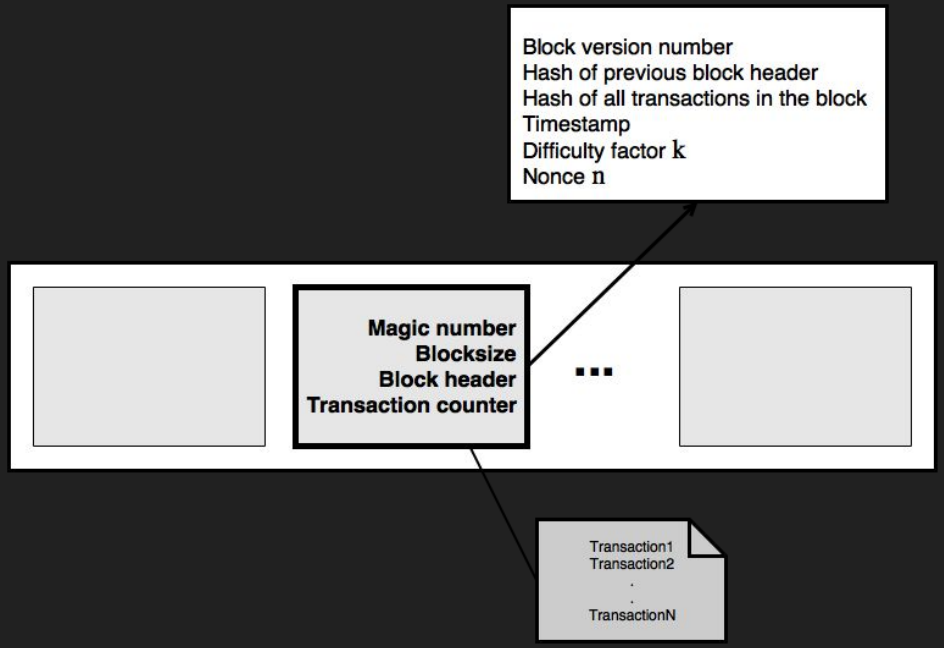
  An EV certificate at Symantec costs $995 / year.

What is a blockchain?

A blockchain is a ledger shared among all computers in a large P2P-network. The blockchain offers data storage which is append only!

This is achieved by making it expensive to add a new block.

In Bitcoin you need to find $SHA2^2(\text{block header} \mid n) < 2^{256 - k}$



Identity on a blockchain

(ID, public key) posted on the blockchain. All subsequent changes to the identity must be signed with the private key.

- The blockchain is distributed, no single point of failure!
- Only one key is valid at a time (the last key in the chain).
- No way for an adversary to replace a keypair without knowing the private key.

```
> pip install blockstack
```

Migrate an identity from a class PKI to a blockchain

The first person to register an identity is considered to be the legitimate owner, similar to how DNS works.

Problem I can hijack Google's identity by posting *(Google, my public key)* to the blockchain.

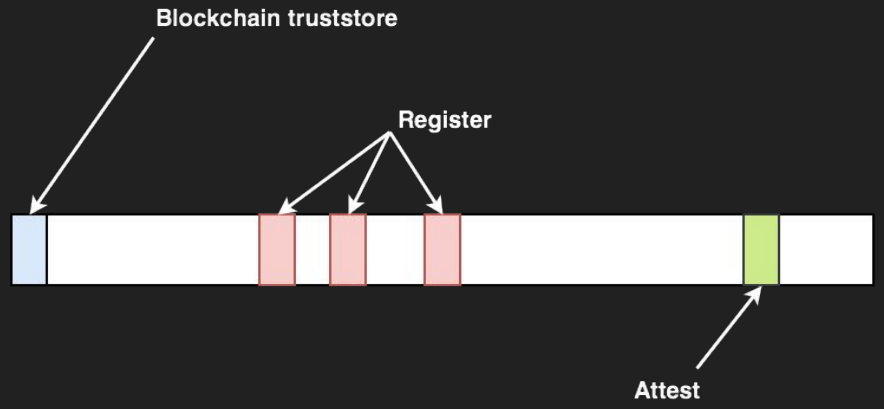
Solution Prove your identity with a certificate and a signature pinned on the blockchain.

Migration process

A blockchain truststore containing all CAs and their public keys is posted at the beginning of the blockchain.

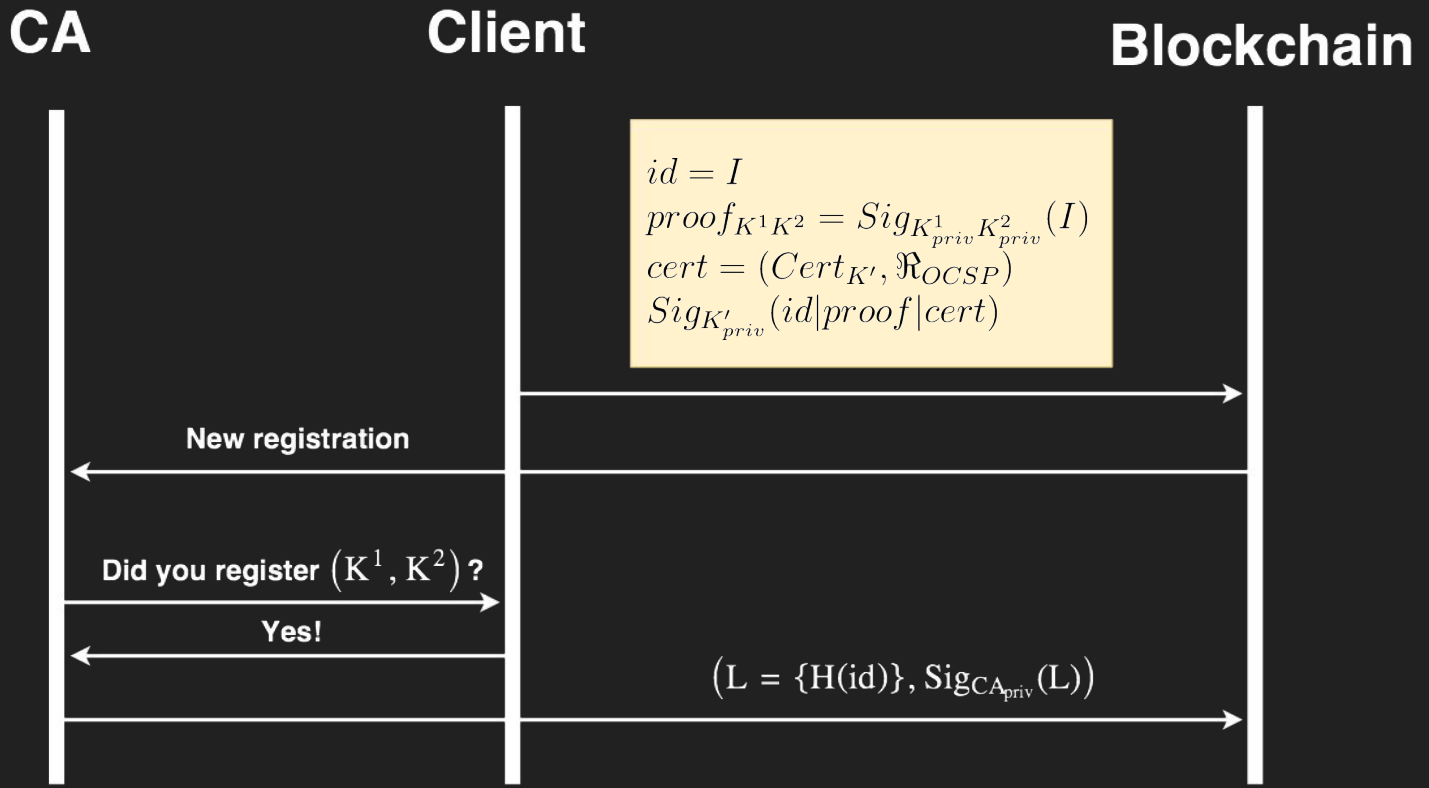
A client registers their identity by pinning a certificate on the blockchain.

Certificates with extended validation needs to be confirmed by a CA.



Make it secure

- Use the timestamp in the block header check for expiration.
- Bundle the certificate with an OCSP response to prove that the certificate is not revoked.
- Sign the transaction with the private key of the certificate to prove ownership.
- Require confirmation from a CA for EV certificates.
- Honour public key pins.
- Check if the CA signature of the certificate is valid using a *blockchain truststore*.


CA
Client
Blockchain
$id = I$
$proof_{K^1K^2} = Sig_{K^1_{priv}K^2_{priv}}(I)$
$cert = (Cert_{K'}, \Re_{OCSP})$
$Sig_{K'_{priv}}(id|proof|cert)$
New registration
Did you register (K^1, K^2) ?
Yes!
$(L = \{H(id)\}, Sig_{CA_{priv}}(L))$

Compare | Product Information | Resources | Why Symantec? | Enterprise Solutions | Renew and Manage Accounts

	Secure Site	Secure Site Pro	Secure Site with EV	Secure Site Pro with EV	Secure Site Wildcard
Price: 1 Year	$399	$995	$995	$1,499	$1,999
Buy Now	BUY	BUY	BUY	BUY	BUY
Warranty	$1,500,000	$1,500,000	$1,750,000	$1,750,000	$1,500,000
Norton SECURED powered by Symantec	✔	✔	✔	✔	✔
ECC: Strongest Security		✔		✔	
Green Address Bar			✔	✔	
Critical Vulnerability Scan		✔	✔	✔	
Blockchain		✔	✔	✔	

Want to renew your SSL Certificate?

RENEW