Blockcerts: Blockchain-based verification of academic credentials

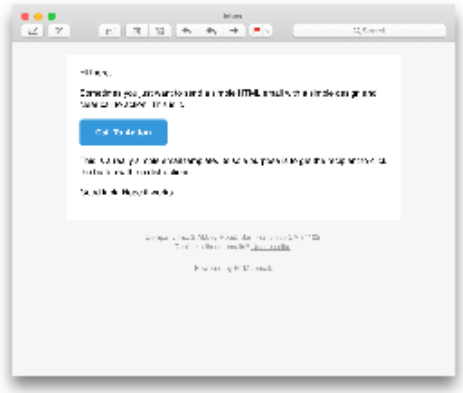
Rujia Li

March 19th, 2018
University of Birmingham

verification of academic credentials

1 show CV

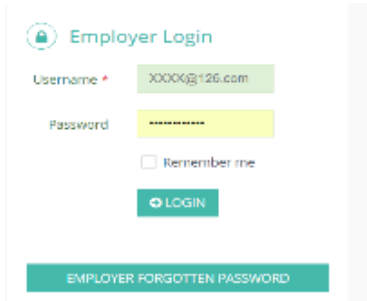
4 confirm requests



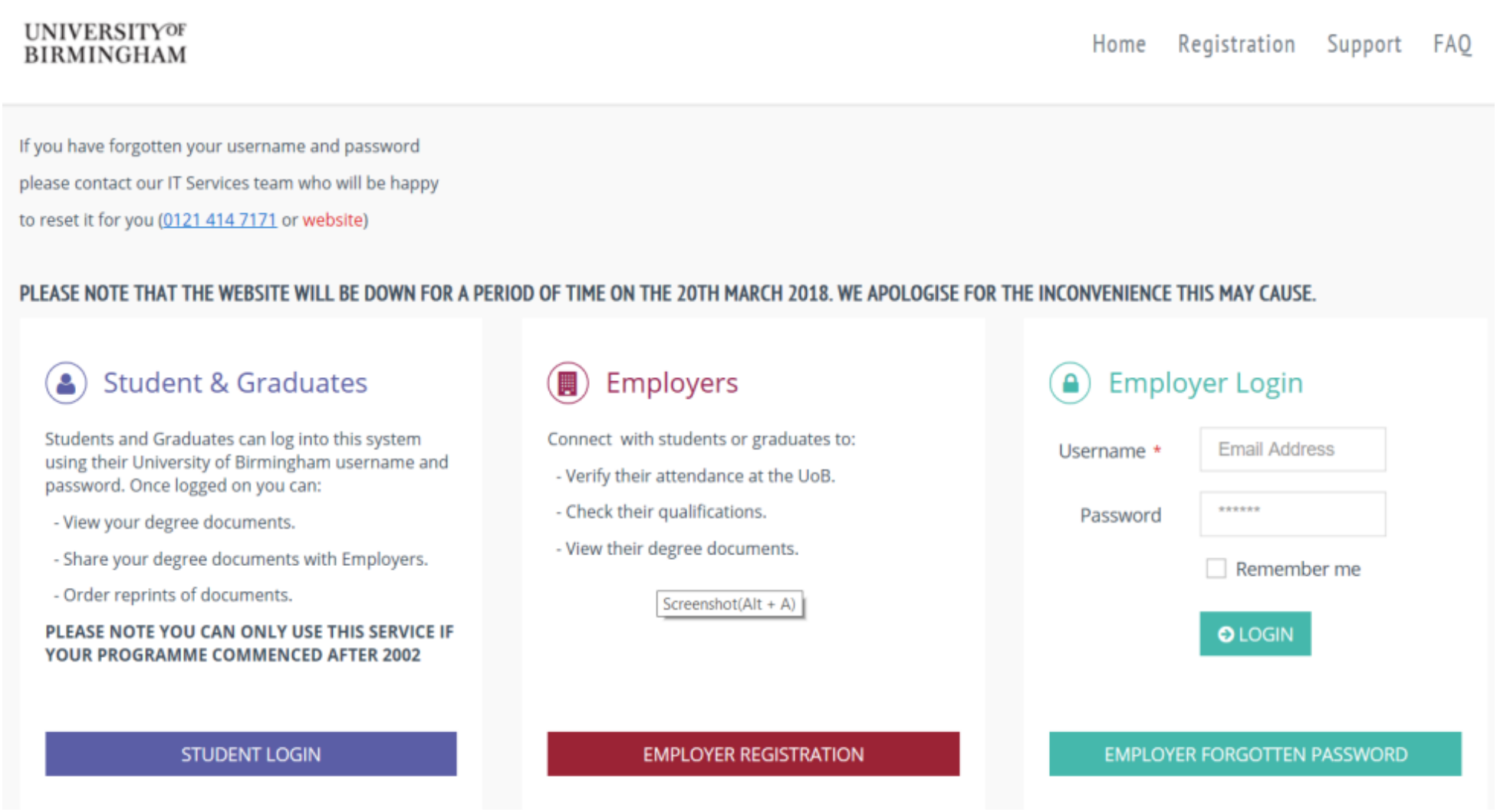
send mails 3

5 feedback

2 create connections



Secure Document website in UOB



Confirmation email

University of Birmingham Secure Document website - Connection Request

SD **Secure Documents <securedocuments@contacts.bham.ac.uk>**
13:37

收件人:

Dear Student,

The following person has emailed you via rxl635@cs.bham.ac.uk and requested access to your documents on the University of Birmingham qualification website.

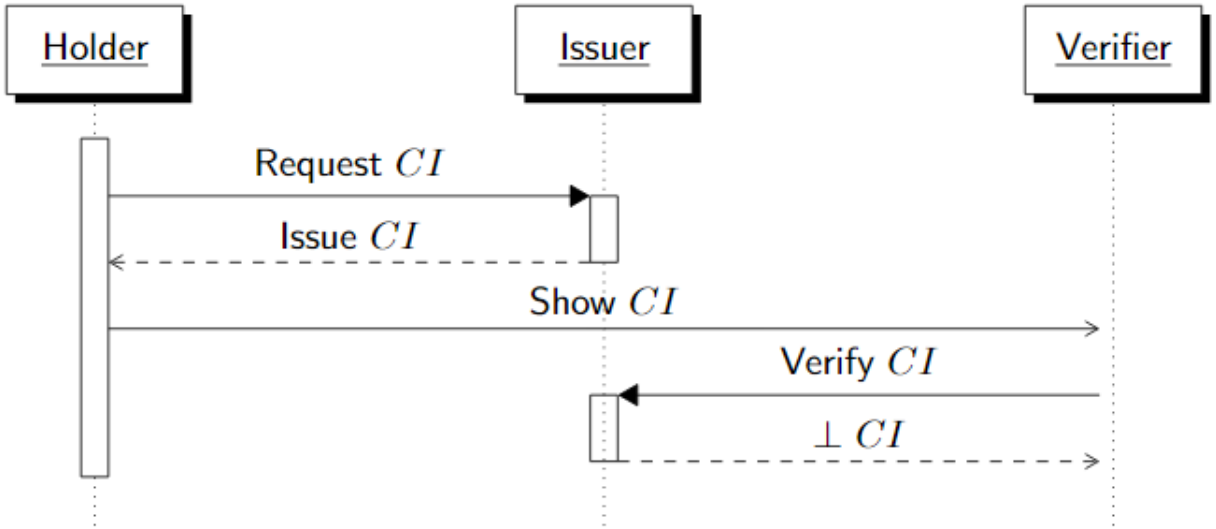
Company: International Business Machines Corporation
Name: IBM Test
Email: rujia1219@126.com

If you would like to go ahead and share your documents with this party please register on the website using your Birmingahm University username and password here: https://verify.bham.ac.uk

This simple process will allow you access to your Secure Documents, allow them to be shared with third parties and allow you to order reprints.

Secure Document Team
Telephone: 0121 414 8669
Email: securedocuments@contacts.bham.ac.uk

PKI-based academic credentials(PACs)



1.A credential holder requests a credential
2.The credential issuer issues the PKI-based credential
3.The credential holder shows the credential to the a verifier
4.The verifier launches a query to issuer to validate the credentials

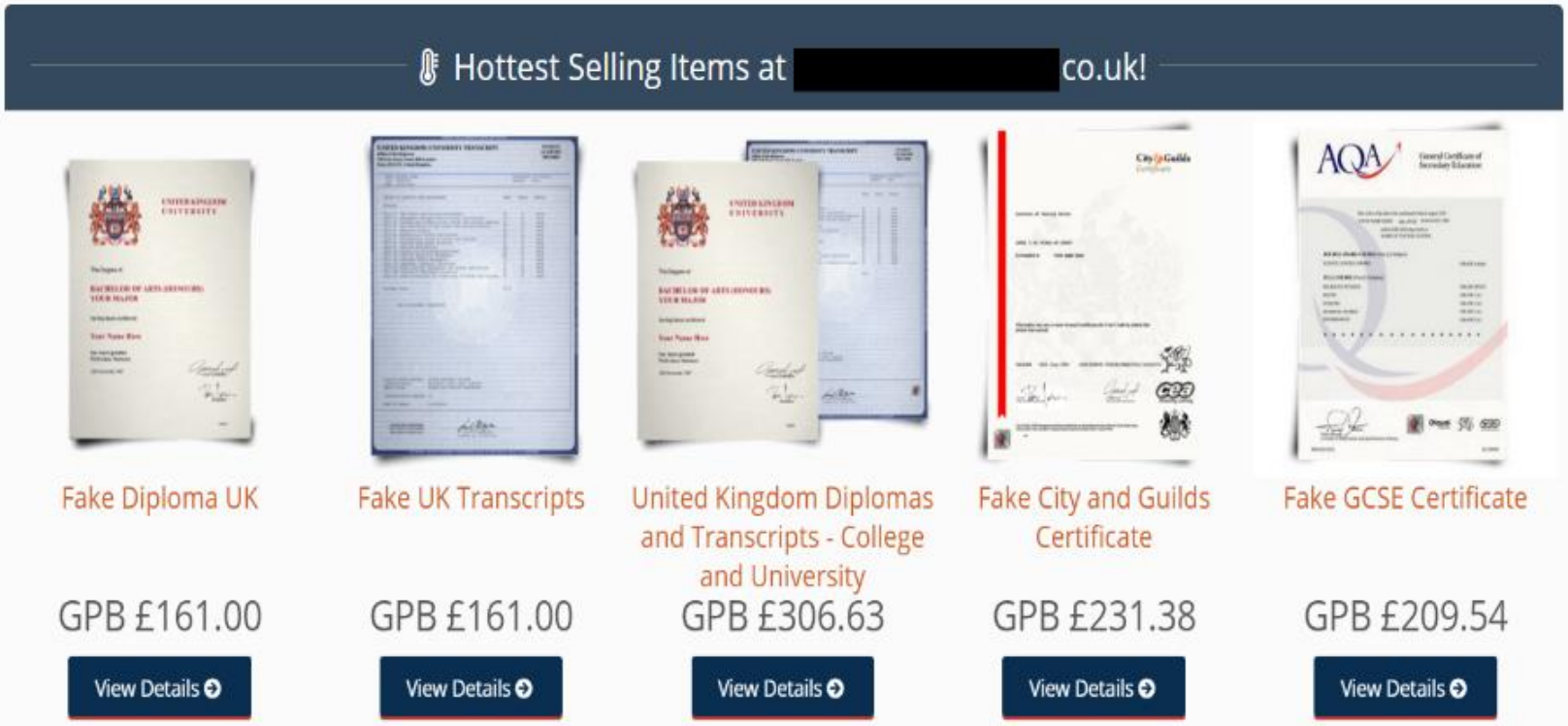
PAC's disadvantages

- **Costly:**
  It is costly to interoperate and collaborate between different business

- **Fragile:**
  Completely dependent on the school. A single point of failure can cause the whole process to fail

- **Centralized:**
  The data is centralized and the verification service must be exposed to outside which increased attack surface

Fake diploma problem

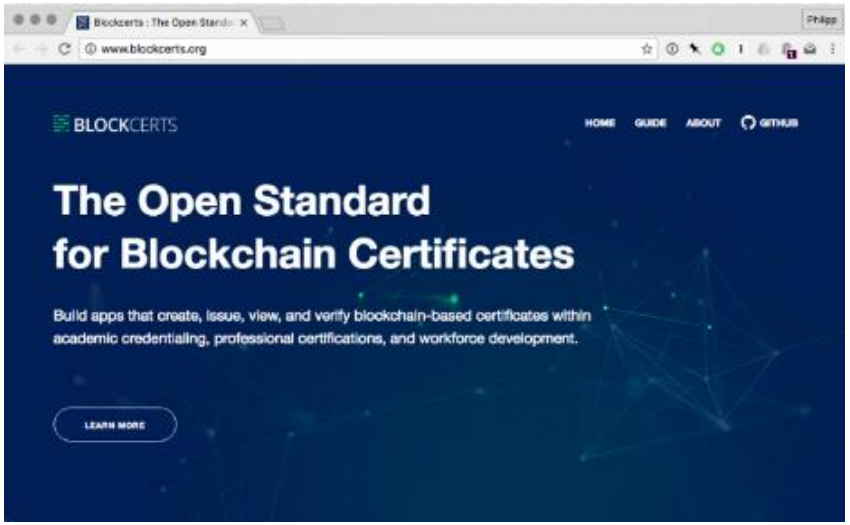


These drawbacks made purchasing a diploma from the internet becomes easy

Blockcerts

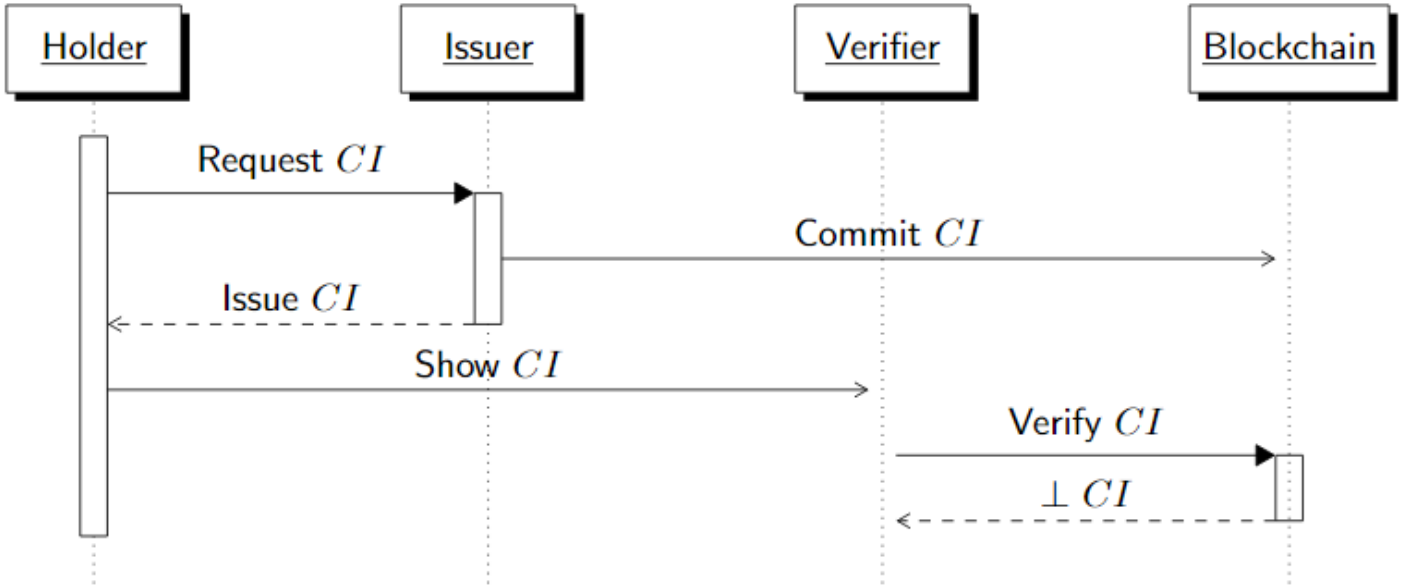
- **MIT Media Lab** released a decentralized credentialing system for academic, professional, and workforce credentialing called Blockchain Certificates in August, 2016 [2]

- In Blockcerts, Bitcoin Blockchain acts as the provider of trust, and credentials are tamper-resistant and verifiable.



2. https://news.bitcoin.com/mit-blockcerts-certification-bitcoin/

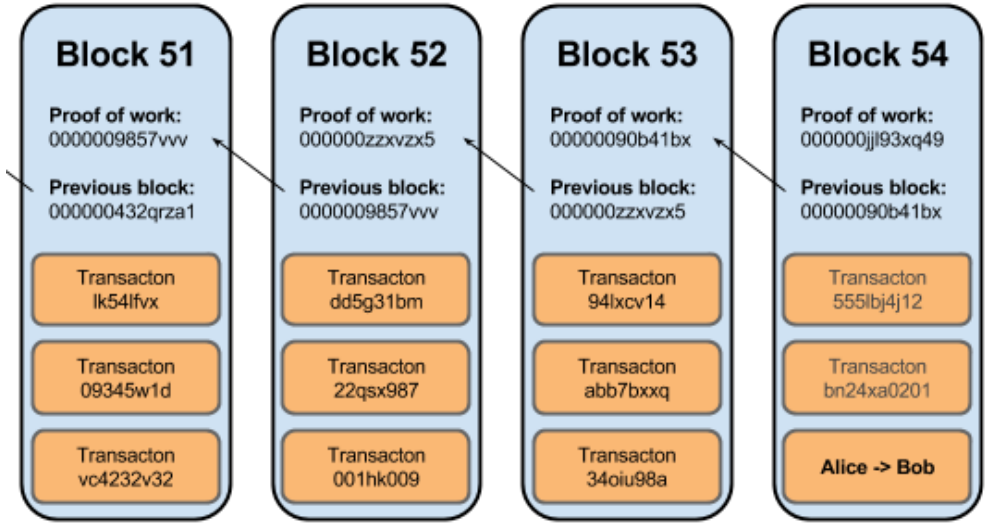
Working diagram of Blockcerts



1. Student requests a credential. 2. Issuer issues the credential CI and commits the credential to the Blockchain. 3. Student provides the credential CI to the verifier (e.g. employer). 4. Verifier accesses the Blockchain to authenticate the certificate

Blockchain

A blockchain is a distributed database that maintains a keep-growing list of ordered records called block [3]. Each block contains a header and a list of transactions TX_i . Each header includes a timestamp T_i, a link to a previous block H_{i-1} and nonce N_i . [4]



The Bitcoin blockchain is cryptographically secured, for every round, the miner need find a random number to meet the computing difficulty D_i , and this progress is called the proof of work (POW) [5].

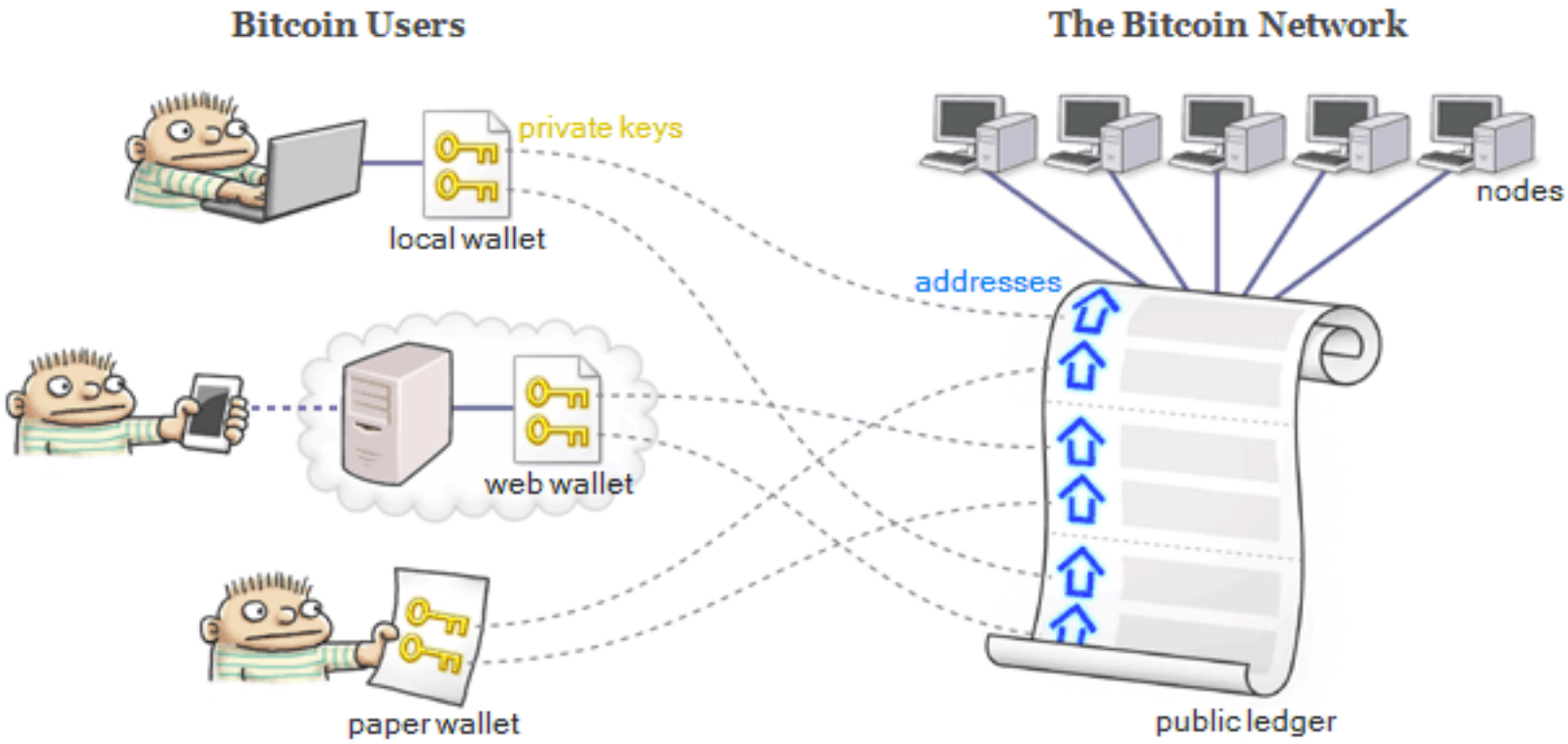
$$f(D_i) > \text{SHA-256}(\text{SHA-256}(H_{i-1} \,||\, T_i \,||\, TX_i \,||\, N_i \,||))$$

3.https://en.wikipedia.org/wiki/Blockchain

4 "Bitcoins In Space". 2017. accessed April 2, 2017, Virgin. https://www.virgin.com/richardbranson/bitcoins-in-space.

5 Alex Biryukov, Dmitry Khovratovich and Ivan Pustogarov. 2014. "Deanonymisation Of Clients In Bitcoin P2P Network".

Blockchain as a bulletin board



The transaction data is decentralized and maintained on every node
No centralized "official" copy exists and no user is "trusted“
Transaction history cannot be changed unless redoing all Proof of Work of all blocks in the chain

Image source: https://busy.org/@jaxteller/how-secure-is-blockchain-discover-below

Commit the hash to Blockchain

Alice sends 5 BTC to Tom with a note “Thank you ”

Alice → Tom 5BTC

Hash

Input Scripts

ScriptSig: 0[] PUSHDATA(72)[3045022100f52838f4959f27bcdb4ced7ec0732db42896744191d467b9dfe7690fbdf
[3045022100d46b1d73392a381e0a0e6c27faee06e3e8321f20574b296ba532a1ca893bf11402205ff8b951451bff7
PUSHDATA1[5221038b1df5a4e05a0ee0a36b5d3a1e44c22ebdb2a0ce57df45650b06b167a85d48d4210379ed77

Output Scripts

DUP HASH160 PUSHDATA(20)[5a1486251ee0bb8022818a2e21da587298db8b6e] EQUALVERIFY CHECKSIG

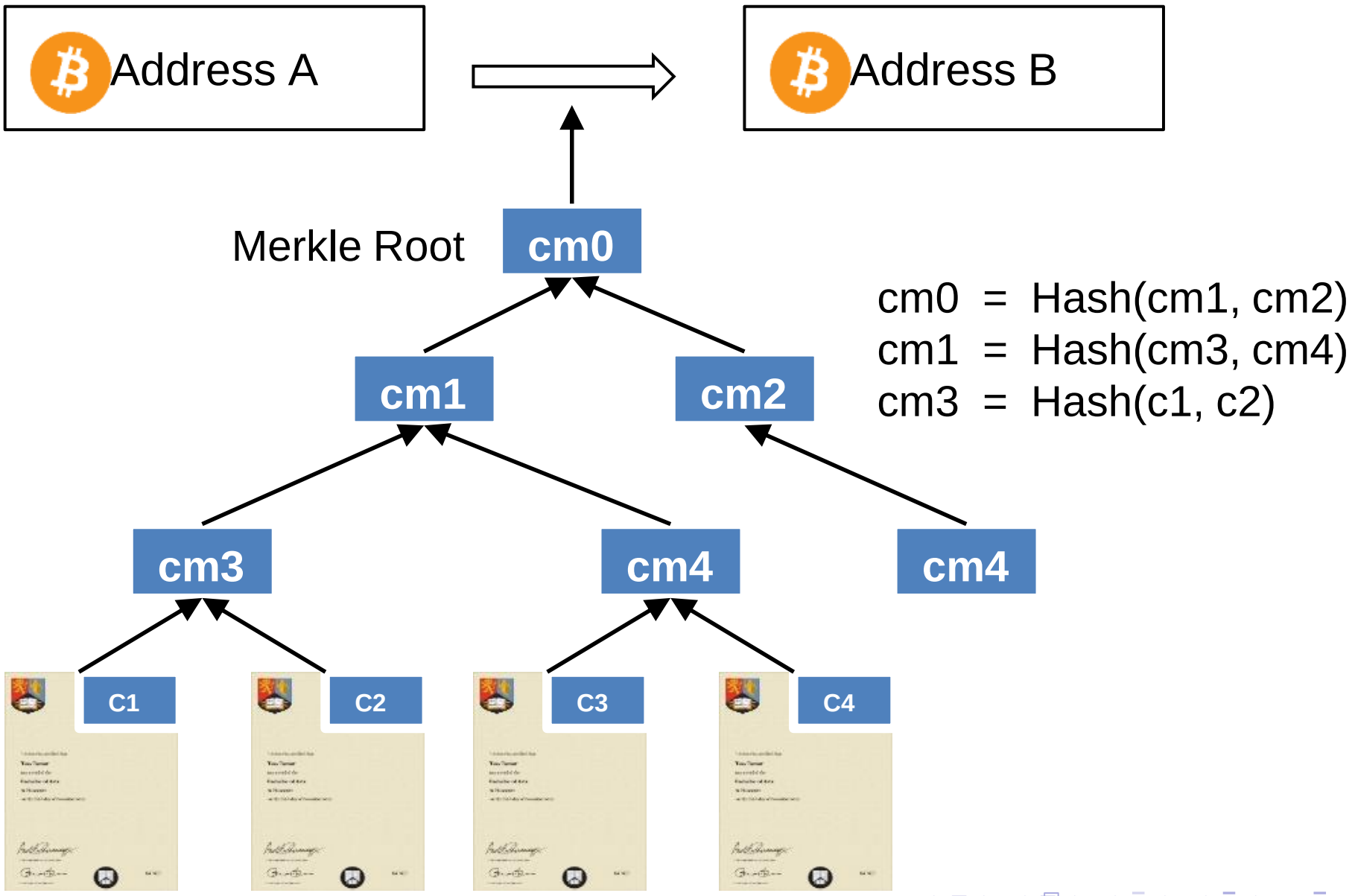
RETURN PUSHDATA(14)[74686973206973206120746573 74]
(decoded) this is a test

Accumulator

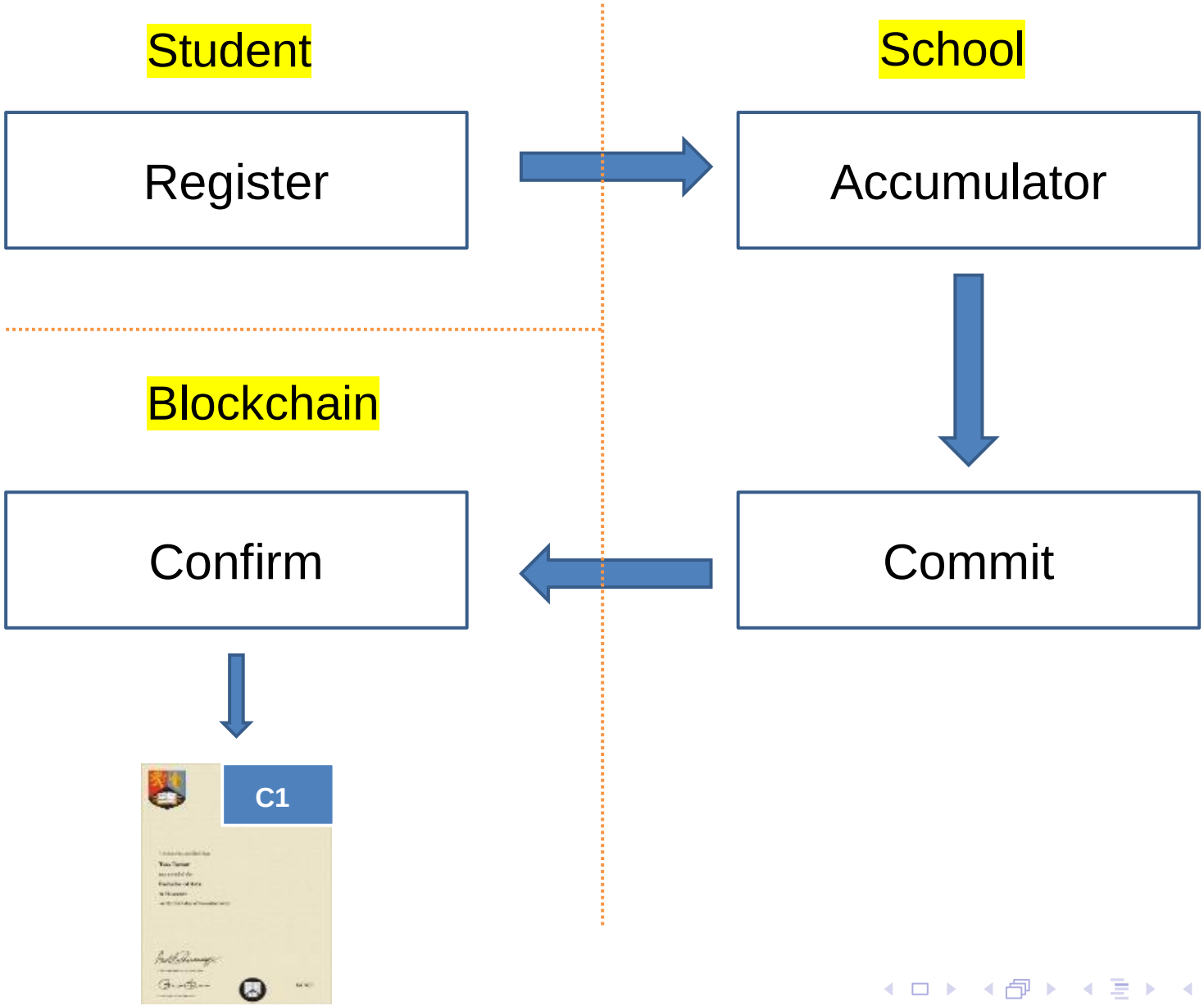
Given an element, an accumulator, and a membership witness, it can be verified that the element is in the accumulated set. A basic accumulator is composed of four polynomial-time algorithms: Gen, Add, MemUpdate and MemVer

+ $Gen(1^k)$ *the credential issuer instantiates the accumulator* a_0 *(representing the empty set), given the security parameter k.*
+ $Add(a_0, y) -> (a', \omega)$ *the credential issuer given the current state of the accumulator* a_0 *and the value(obtained by credential holder) to be added* y*, then returns the new state of the accumulator* a' *and its corresponding witness* ω*.*
+ $MemUpdate(\omega, y') -> (\omega)$ *credential issuer employ the current state of a witness* ω *and the new value* y' *being added to the accumulator, and returns an updated witness* ω' *.*
+ $MemVer(a', y, \omega) -> \{0, 1\}$ *credential verifier verifies the membership of* y *in the accumulator* a' *using its witness* ω*, and returns 1 if y appears to be in a, and 0 otherwise.*

Commit the Merkle root to Blockchain



Issuing the Blockcerts



Standard blockchain certificate


```
{
  - badge: {
        created: "2017-01-01",
        description: "good",
        expires: "2100-01-01",
      + fileClaim: { … },
        id: https://example.org/robotics-badge.json,
      + identityClaim: { … },
        image: "good",
      + issuer: { … },
        name: "Bachelor of Arts",
      + revocationClaim: { … },
        type: "Certifacte"
    },
  + context: [ … ],
    id: "1a08a38afe7f4a848d8f6e7609350814",
    issuedOn: "2017-08-15 00:20:26",
  + recipient: { … },
  - signature: {
      + anchors: [ … ],
        context: https://w3id.org/chainpoint/v2,
        merkleRoot: "1cf7ec6048c93cb3710b274a714ff5e7312b496f7cf79bd27881feed69c122eb",
      - proof: [
          - {
                left: "68a7d9a8e1f47d23e28e57e15fdbbc3206764363a93db32705aa8ceddfb96116"
            },
          - {
                right: "7a1337ce6ce66b6114b7828f82d8a20477d9db37cef76c7841f2b10365f45508"
            }
        ],
        targetHash: "4ec37c5ab0595ca0ba0f5cb80526ed7dbee0da13636ee3c15e47dc953089d4e9",
      + typelist: [ … ]
    },
    type: "badgeClass",
  + verification: { … }
}
```

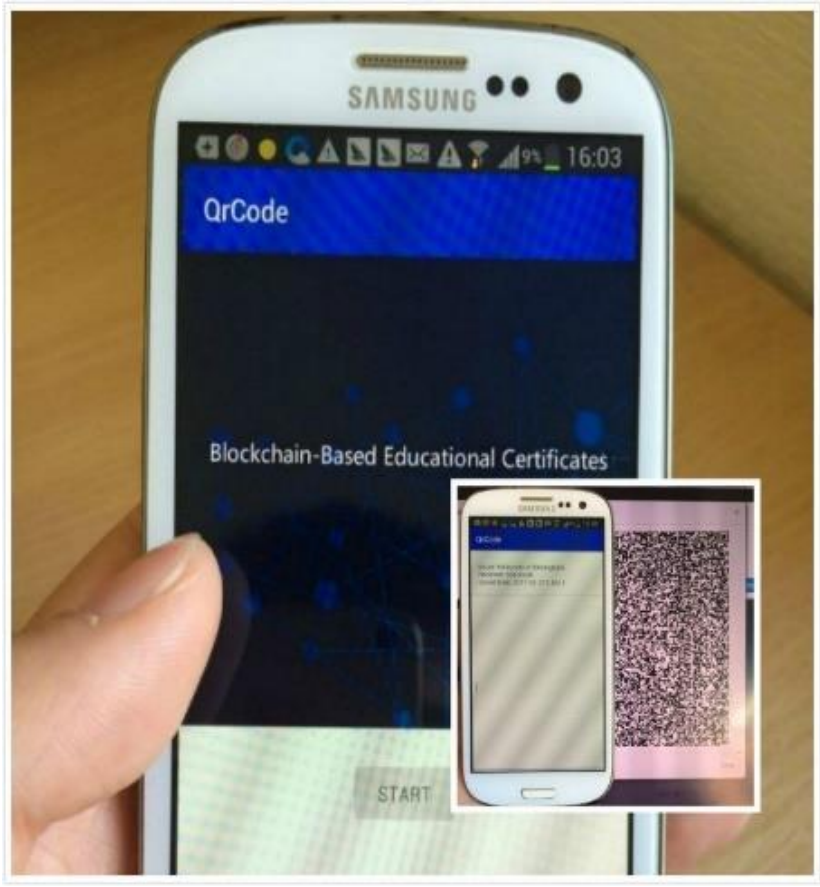

Certificate

Receipt

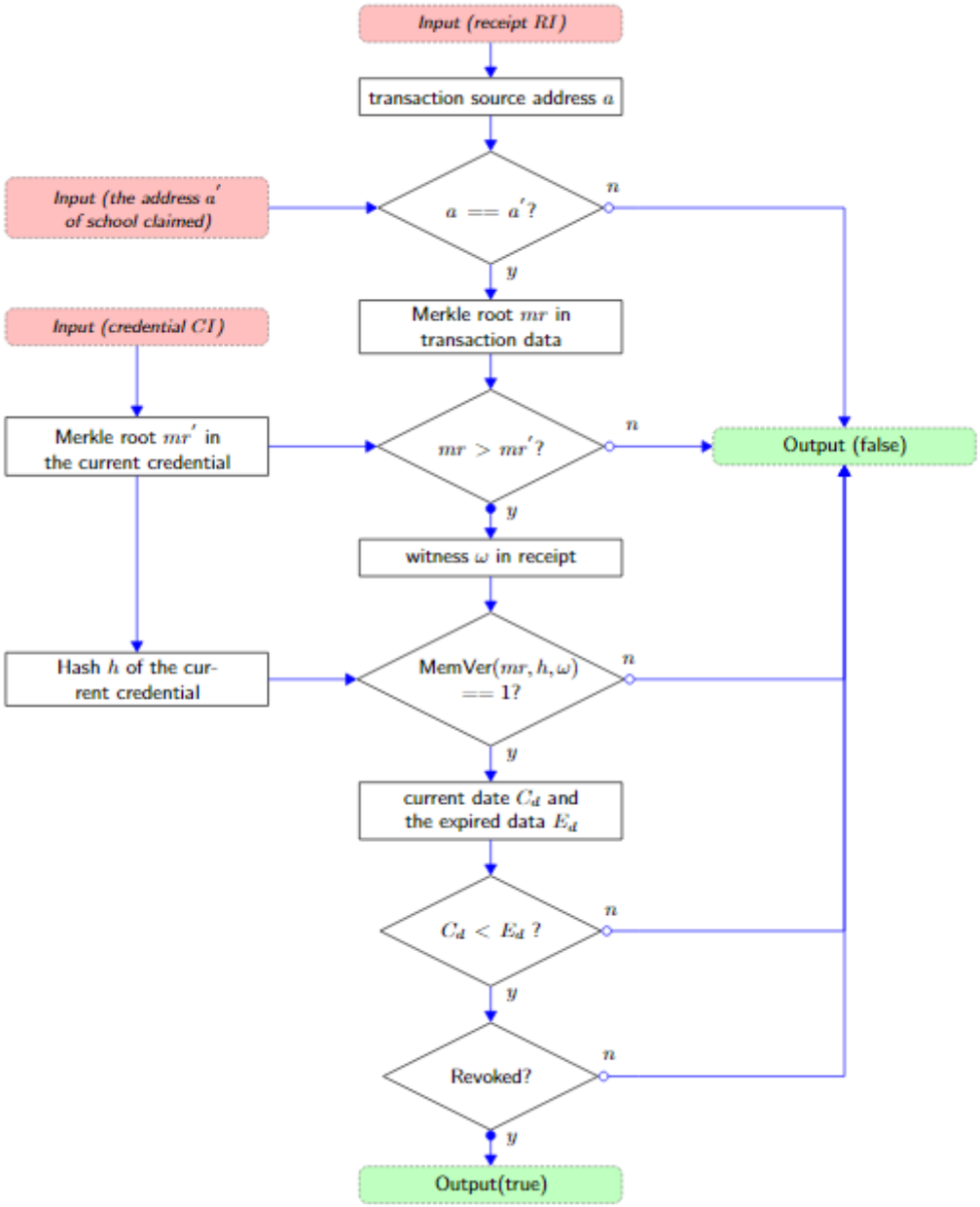
Blockcerts verification

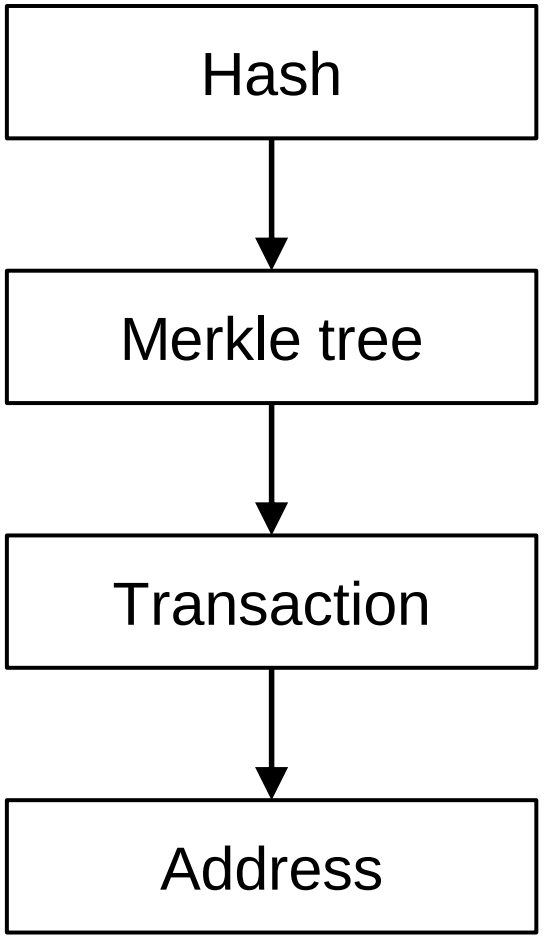
Launch the app, start scanning the QR code to verify

QR Code for Graduate/Employer

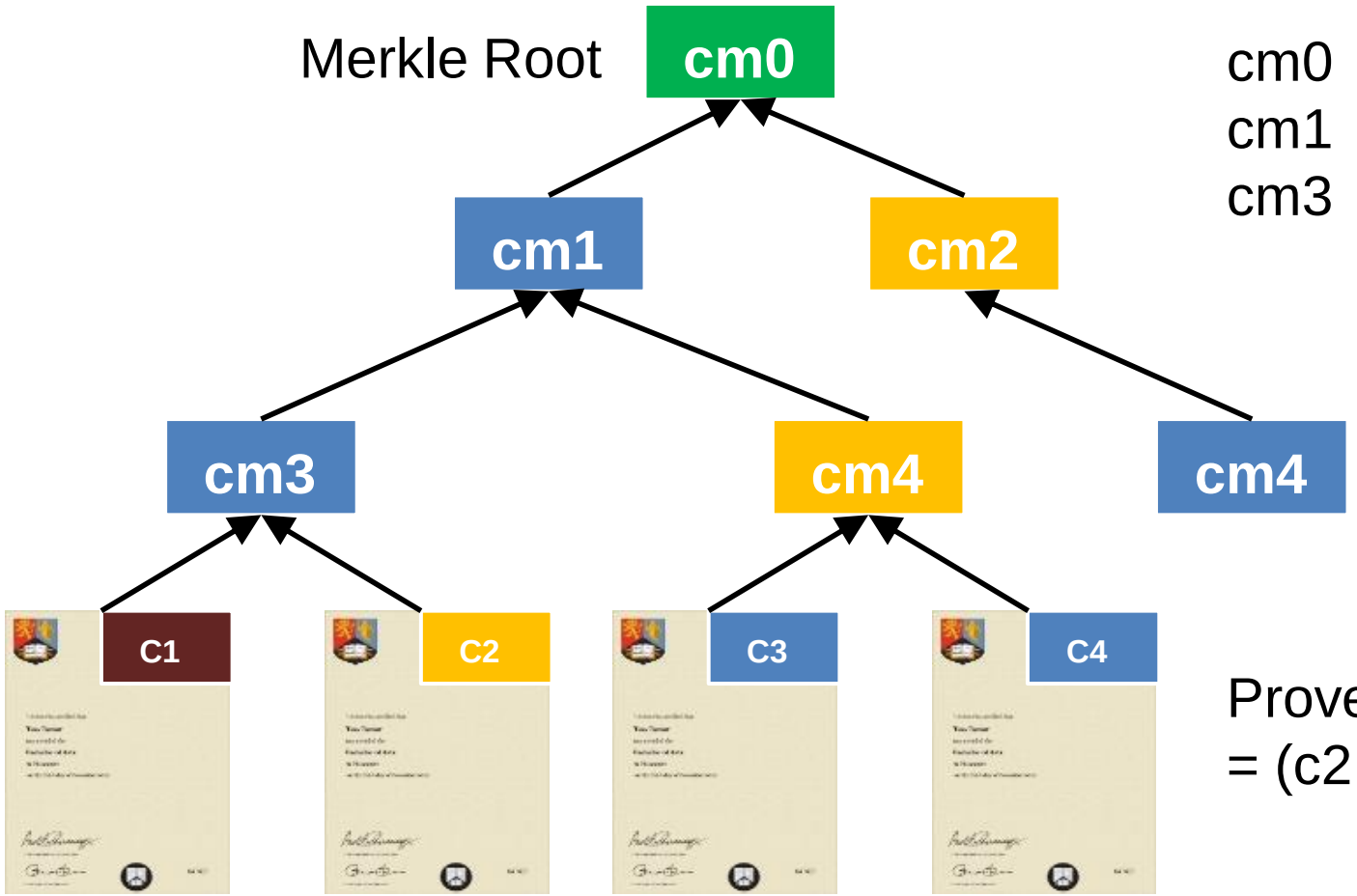


Blockcerts verification





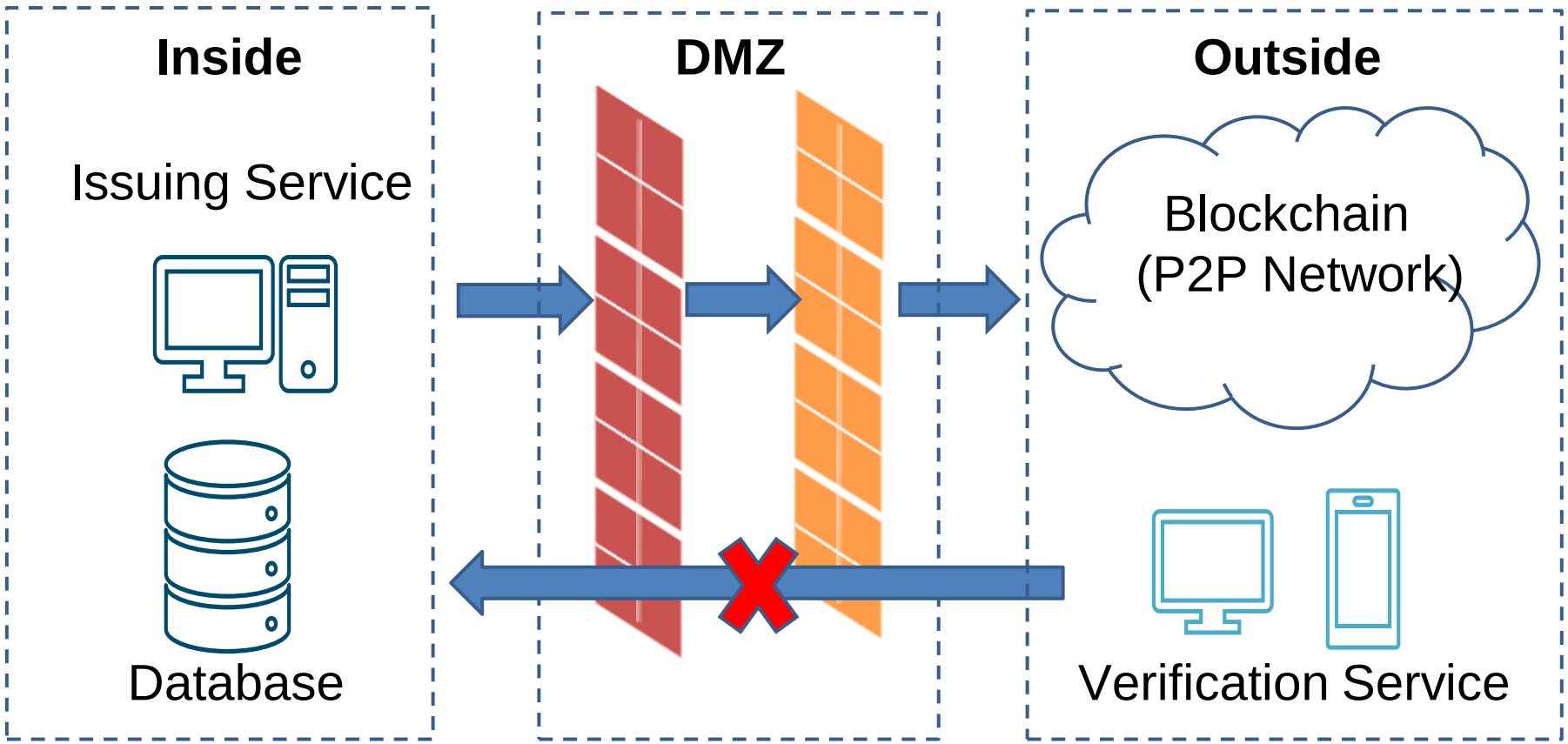
Merkle presence proof



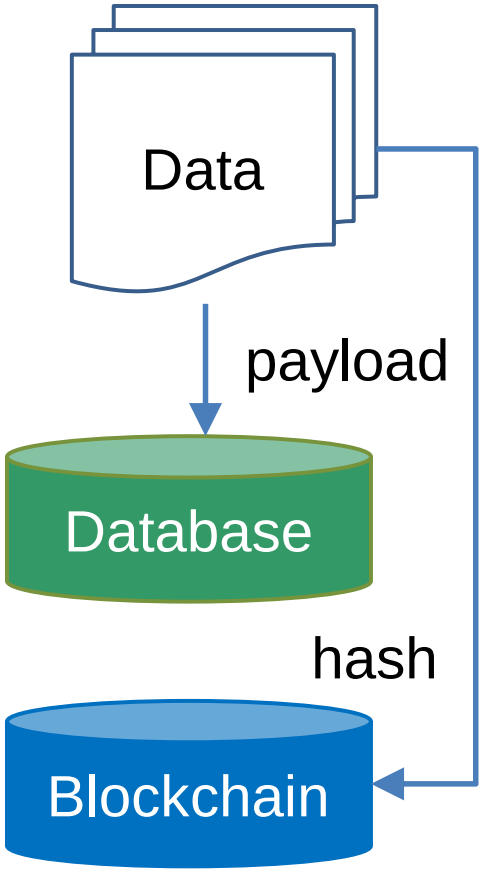
cm0 = Hash(cm1, cm2)
cm1 = Hash(c3, cm4)
cm3 = Hash(c1, c2)

Prove_presence(cm0, c1) = (c2, cm4, cm2).

More secure(network) ?



More secure(data) ?



The data is stored in the school, even the Blockchain failed, there is no data leakage risk.

If anyone modified the existing records at the service level it will be detected.

Any internal staff manipulating database will be found.

My current research

- Prove security of a protocol as stand-alone is easy.
  (single execution, no other parties).

Need:

- A general framework for representing security concerns and requirements for protocols

- A general composition operation that:
  –Captures realistic situations in multi-protocol systems
  –Preserves security

Thank you