Blockchain-based Transcript Inquiry and Verification System

Author: ZHIHAO, LIANG  Supervisor: Bertrand M.T. Lin

梁志豪  林妙聪

National Chiao Tung University
Department of Information Management and Finance
Outline

Motivation and Problem Statement
Background
System Architecture
Results and discussion
References
Motivation and Problem Statement:

Outline
Motivation and Problem Statement
Background
System Architecture
Results and discussion
References
Motivation and Problem Statement:

Many applications have been developed on the basis of Blockchain. In this paper, we are going to propose a new application scenario of Blockchain. The application we are going to build is a Grade Inquiry and Verification System (GIVS) based on Blockchain. Traditionally, examinees’ transcripts are delivered by postal services or by an official website. In these ways, transcripts are vulnerable to attacks. Transcripts may get lost or be tampered by adversaries. The server providing grade inquiry services may suffer from denial-of-service attack when a large number of students are querying at the same time. Single point of failure is also a threat to servers of the system. Moreover, these systems are lack of the mechanism to detect tampered transcripts. In contrast, our GIVS system is free from the problems described above. GIVS has inherited many superior characteristics from Blockchain. It outperforms traditional grade querying systems in many aspects, such as reliability, privacy and security. GIVS also make it possible to access and verify transcripts easily and at lower cost.
Motivation and Problem Statement:

Traditional Grade Inquiring and Verifying System (GIVS)

- Be tampered by adversaries.
- Denial-of-service attack
- Single point of failure
- Lack of mechanism to detect tampered transcripts.

GIVS

- Outperforms traditional grade querying system
- Reliability, privacy, and security
- Easily and at lower cost.
Getting your results

Your Test Report Form will be available 13 days after you complete the test. You will receive one copy of the form, or two copies if you are applying to Citizenship and Immigration Canada (CIC). You can arrange for your test centre to post this to you, or you can pick it up in person.

Viewing your results online

Your results will also be available to view online for 28 days. (This should not be used as an official confirmation of your performance).

Your IELTS test centre will provide you with a link to your results or you can view them via one of these websites:

- British Council network of test centres
- IDP IELTS Australia network of test centres
- IELTS USA network of test centres

You will need your passport or ID number (the same number you used when you registered for the test) and your candidate number.

If you have any questions or problems, contact your test centre.

Sending results to nominated organisations

When you book your test, you can nominate up to five organisations to be sent your IELTS test results on your behalf. This service is free of charge. Results can be sent to further nominated organisations for a small administration fee.

If your centre has closed, you can ask for your IELTS result to be sent to your nominated organisations by organisations (by filling in the Application for additional TRFs (from closed centres) form. This service is available for up to two years from the date of your IELTS test.
Please note that this online results service is only available through selected British Council managed test centres.

The system will not display results for more than 40 days after the test date.

Search for your Result

- Test Date: *
- Date of Birth: *
- Identification Document Number: *
- Candidate Number: *

Find

Your grades will be kept on this Query Website for only 40 days!
Background:

Abstract
Outline
Motivation and Problem Statement
Background
System Architecture
Results and discussion
References
I. Hash function
II. Blockchain
III. Bitcoin Null Data Transaction
IV. Hierarchical deterministic wallet
A hash function is any function that can be used to map data of \textit{arbitrary size} to data of \textit{fixed size}. 
A blockchain is a continuously growing list of records, called blocks, which are linked and secured using cryptography. Each block typically contains a hash pointer as a link to a previous block, a timestamp and transaction data. By design, blockchains are inherently resistant to modification of the data.
Null data transaction type relayed and mined by default in Bitcoin that adds arbitrary data to a provably unspendable pubkey script.
Hierarchical deterministic wallet:

Keys:
- Private key
- Public Key
System Architecture:

Outline
Motivation and Problem Statement
Background
System Architecture
Results and discussion
References
Operating Procedures of GIVS system:

1. Doing test
2. Broadcasting score
3. Access score
4. Providing account information like: blockchain address
5. Access & verifying score info
Account Management:

Examinees will receive:

- Round Number
- Examinee ID
- His/Her Keys (private key and public key)

Master public key is known to all!
Take IELTS as an example. We have 5 pieces of information to record:

- Listening
- Speaking
- Reading
- Writing
- Total Grade

Simple transcript
Transcripts Broadcasting:

1) Each examinee needs to prepare enough of transaction fees for his/her own payment address of blockchain.
2) Examination Institution builds transactions with transcripts attached to them, and broadcast them to blockchain.
3) The input and output address are examinee’s payment address
Security issue:

Examinees may make similar transactions themselves to fabricate new transcripts for their interest.

Solution:

- Pay to script hash
- 2 of 3 Multi-signature

The real address of the examinee is:

\[
\text{Examinee\_Addr} = \text{RIPEMD160( 2 MasterPubKey RoundPubKey ExamineePubKey 3 OP\_CHECKMULTISIG )}
\]

Only the one who holds at least 2 of Master Private Key, Round Private Key, and Examinee Private Key, can make the transaction.
Revised transaction:

\[ \text{Examinee}_\text{Addr} = \text{RIPEMD160( 2 MasterPubKey } \\
\text{RoundPubKey ExamineePubKey 3 } \\
\text{OP\_CHECKMULTISIG )} \]
Revocation:

Transcripts have to be revoked if they are expired.

Mechanism:

Valid: Round Key Address has UXTO
Expired: Round Key Address has no UXTO

Note: UXTO: Unspent Transaction Output
Revocation(.cont) :

Transcripts have to be revoked if they are expired.

Mechanism:

Valid: Round Key Address has UXTO
Expired: Round Key Address has no UXTO

Note: UXTO: Unspent Transaction Output
Enquiry on Results:

Block 0 — Block 1 — Block 2 — Block 3 — Block 4

Old Transcript — Invalid
New Transcript — Valid

Block height is maximum!
Privacy:

To protect the privacy of examinees, examination institution need to encrypt transcripts before broadcasting them to network.

Encryption:
\[ C = AES256\_encrypt( M, hash256(ExamineePriKey) ) \]

Decryption:
\[ M = AES256\_decrypt( C, hash256(ExamineePriKey) ) \]

However, this will lead to another problem:
Privacy(.cont 1):

When the third party has obtained $C$ from Examinee_Addr, it’s hard to get $M$ without knowing the private key of the examinee.

If the third party invite the examinee to decrypt the cyphertext $C$, he is unable to be sure that the examinee is honest.
Solution:

Attach hash of the transcript to the null data transaction before sending it to the network.

Denote $T$ as a random number of a reasonable size.

Then, the message of Null Data transaction is:

$$\text{Null_data_Message} = \text{OP}_\text{RETURN} < C \parallel \text{hash160}(M \parallel T) >$$

$\text{Hash160()}$ must be a hash function with hiding property.

Note:

Encryption:

$$C = \text{AES256}_{\text{encrypt}}(M \parallel T, \text{hash256(ExamineePriKey)})$$

Decryption:

$$M \parallel T = \text{AES256}_{\text{decrypt}}(C, \text{hash256(ExamineePriKey)})$$
Authentication:

[The first step]
The third party receives the public key from the examinee. And verify that it’s a descendant of Master public key.

[The second step]
[1] Send random number "R"
[2] Return ECDSA of "R"
[3] Check validity of ECDSA
Verifying transcript:

The third party invites examinee to decrypt $C$, and receive $M' || T'$ from him.
Then the third party does the following to check the validity of the reported transcript $M'$

\[
\text{Validity} = \text{is\_equal}( \text{hash160}(M' || T'), \text{hash160}(M || T) )
\]

If validity is true, then we know that $M = M'$, and the transcript is valid.
Results:

Outline
Motivation and Problem Statement
Background
System Architecture
Results and discussion
References
Implementation:

Dependency relation
## Broadcasting time complexity:

<table>
<thead>
<tr>
<th>Number of examinees</th>
<th>Time (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13.28</td>
</tr>
<tr>
<td>10</td>
<td>8.53</td>
</tr>
<tr>
<td>1000</td>
<td>33.71</td>
</tr>
<tr>
<td>10000</td>
<td>95.69</td>
</tr>
<tr>
<td>no. of bits ((b))</td>
<td>hash space size ((2^b))</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td></td>
<td>(p = 10^{-18})</td>
</tr>
<tr>
<td>32</td>
<td>2.9×10³</td>
</tr>
<tr>
<td>(40)</td>
<td>47</td>
</tr>
<tr>
<td>(48)</td>
<td>24</td>
</tr>
<tr>
<td>64</td>
<td>6.1</td>
</tr>
<tr>
<td>(96)</td>
<td>4.0×10⁸</td>
</tr>
<tr>
<td>128</td>
<td>2.6×10¹³</td>
</tr>
<tr>
<td>(192)</td>
<td>3.5×10²¹</td>
</tr>
<tr>
<td>256</td>
<td>1.5×10³⁴</td>
</tr>
<tr>
<td>(384)</td>
<td>2.8×10⁵³</td>
</tr>
<tr>
<td>512</td>
<td>5.2×10⁷⁵</td>
</tr>
</tbody>
</table>

**Collision probability table**
Results and discussion:

Strengths of GIVS:

- Open to everyone (Adaptability)
- Tamper-proof (Immutable)
- Reliable, stable
- Protect the privacy of users
- Simple to check the validity of transcripts
- Convenient and easy to use
- Lower cost than traditional systems
- Information will be kept forever
Issues and future work:

Defects inherited from Bitcoin:

- Throughput: Only 1 Megabytes for each block
- Latency: New blocks are generated every 10 minutes
References:


Thanks for your time!

Q & A