Blockchain Based Smart Contract Secure Charity System

Group 7
Zhehao Fan, Tinghui Wu,
Rundong Liang, Kaiyi Chen

Contents:

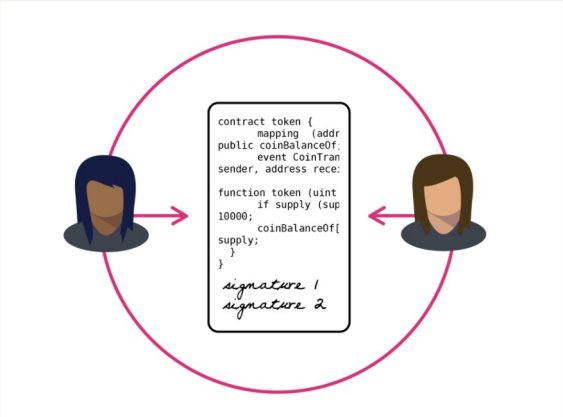
Introduction

Smart Contract ：

A smart contract is a self-executing computer program that can automatically execute the code within it without any human intervention

The advantages of smart contract:

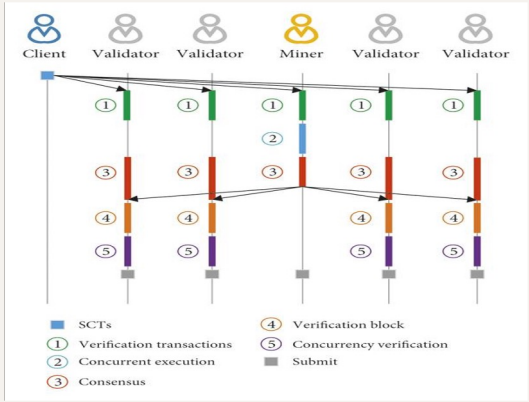
1. Automated execution
2. Decentralized
3. Tamper-proof

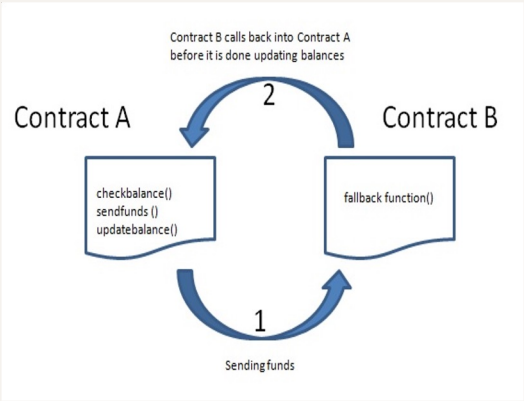


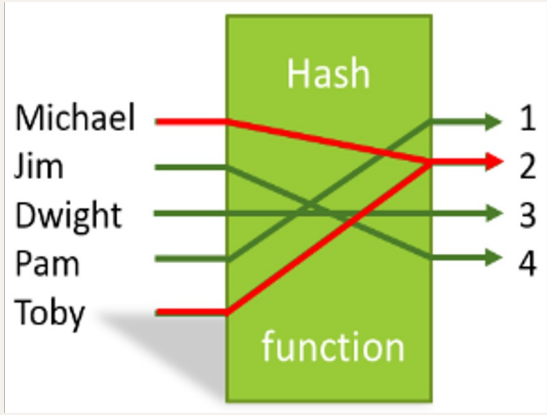
The Implementation Challenges

1. Scalability problem: inability to meet the needs of highly concurrent scenarios

2. Security issues: vulnerability attacks, code errors, malicious contracts.



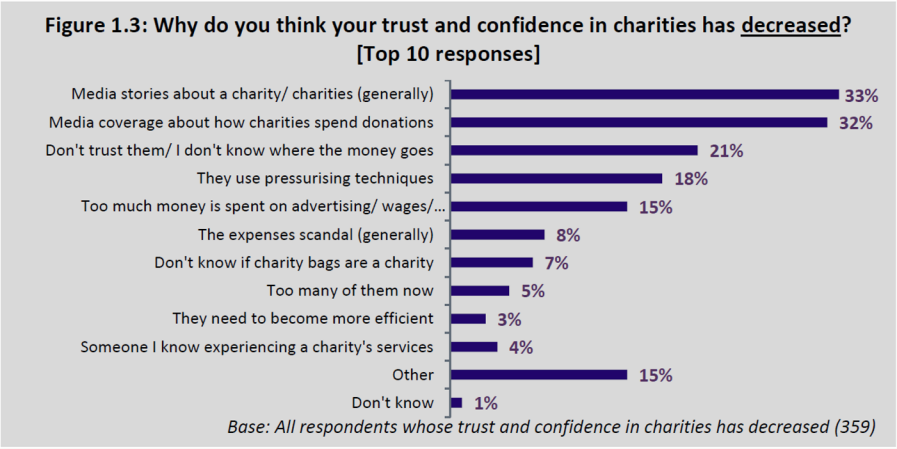




Problem Statement

Traditional charities often suffer from lack of trust

1. Non-transparent financial management
2. Corruption
3. Information asymmetry



Public Trust 1: How can we rebuild public trust in charities? - Charity Commission (blog.gov.uk)

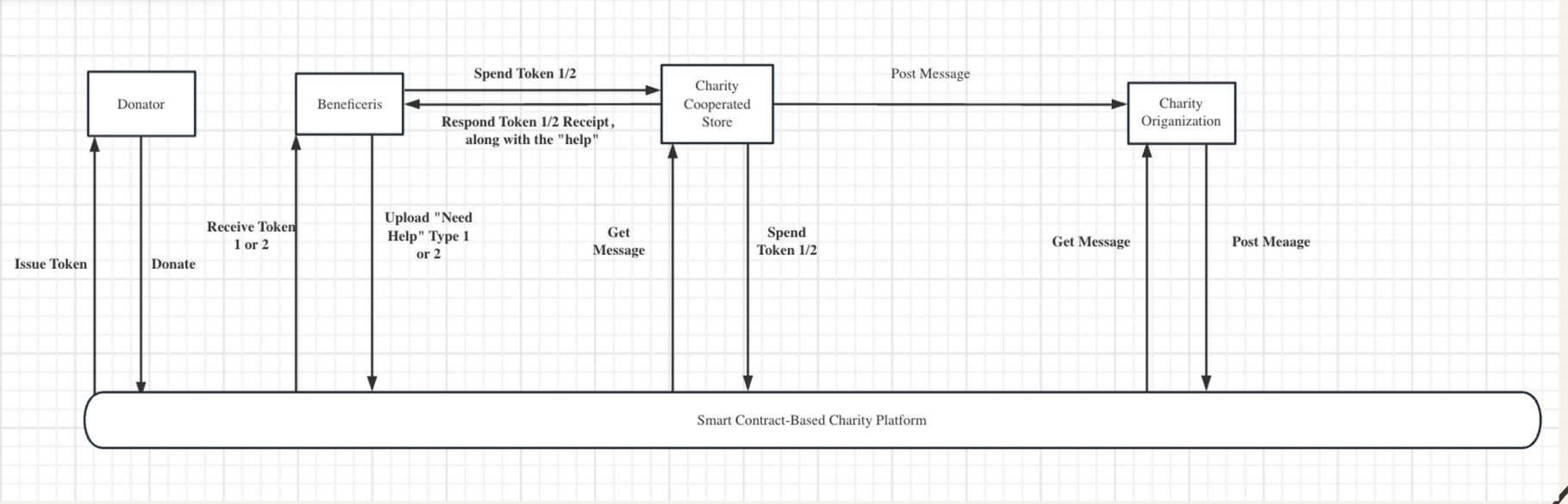
Smart Contract in Charity System

1. Smart contracts enable transparency and fairness in donations to charities.

2. The tamper-evident nature of smart contracts provides a higher level of trust for donors.

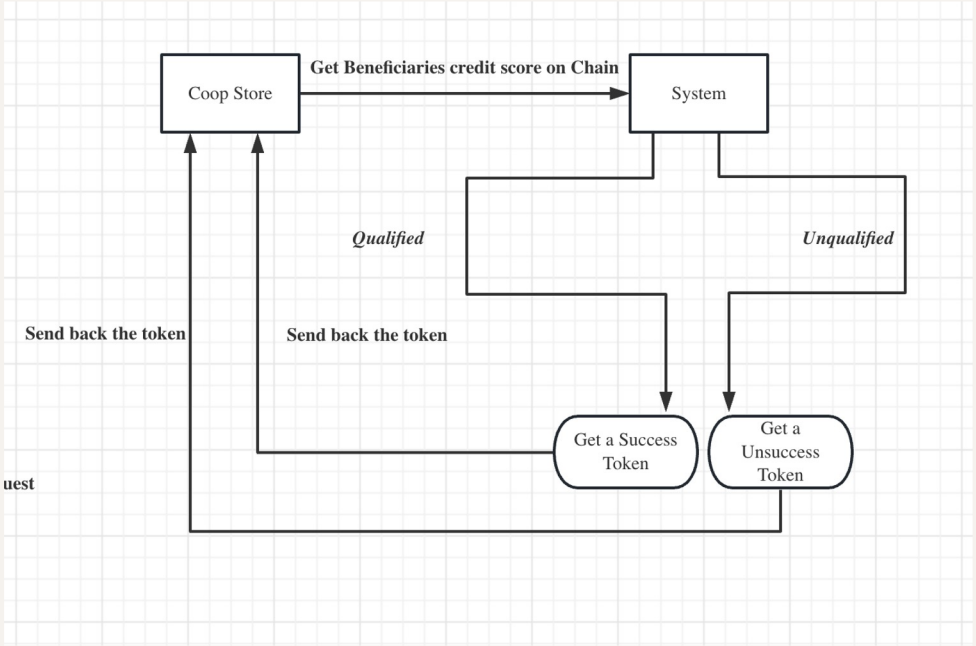
3. Smart contracts can effectively manage and track the flow of funds and expenditures of charities

System Model



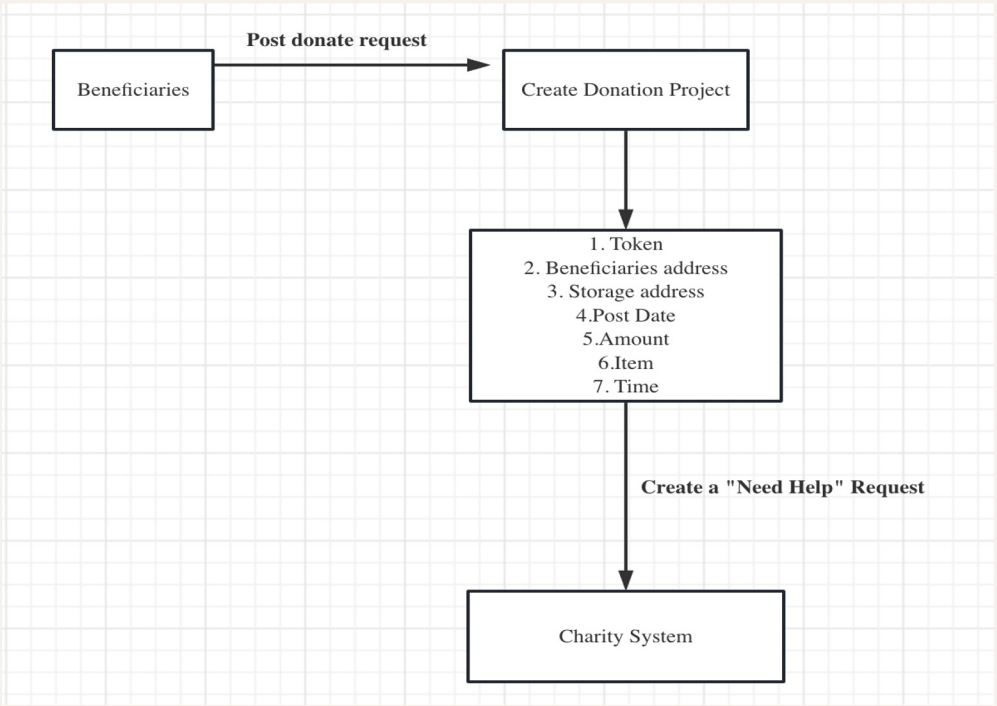
Flow Chart

Beneficiaries get donated:



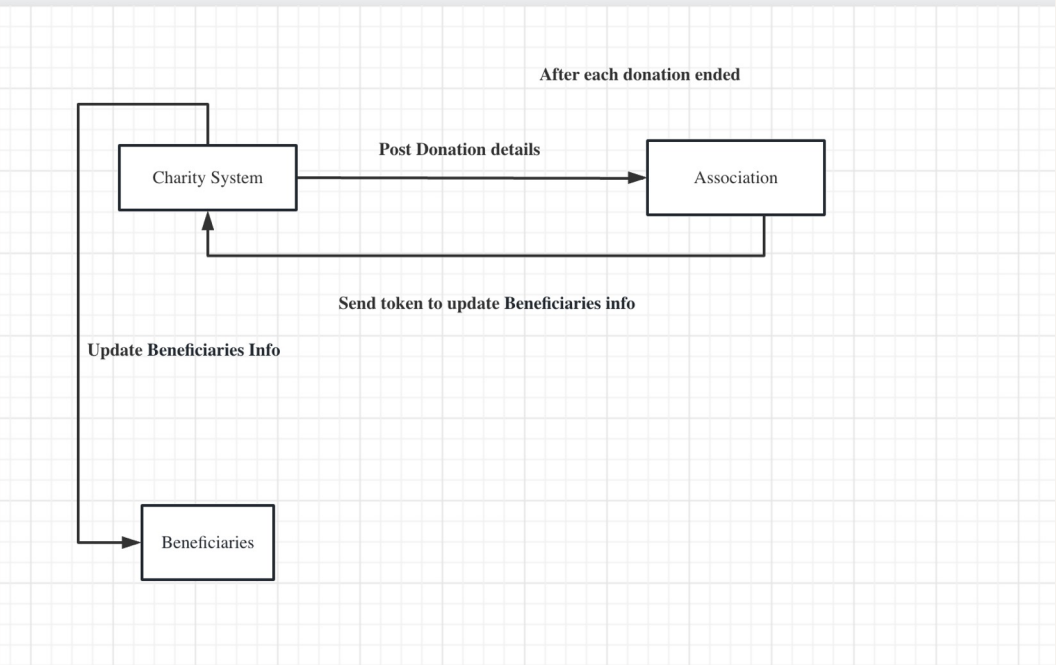
Flow Chart

Beneficiaries post donate request:

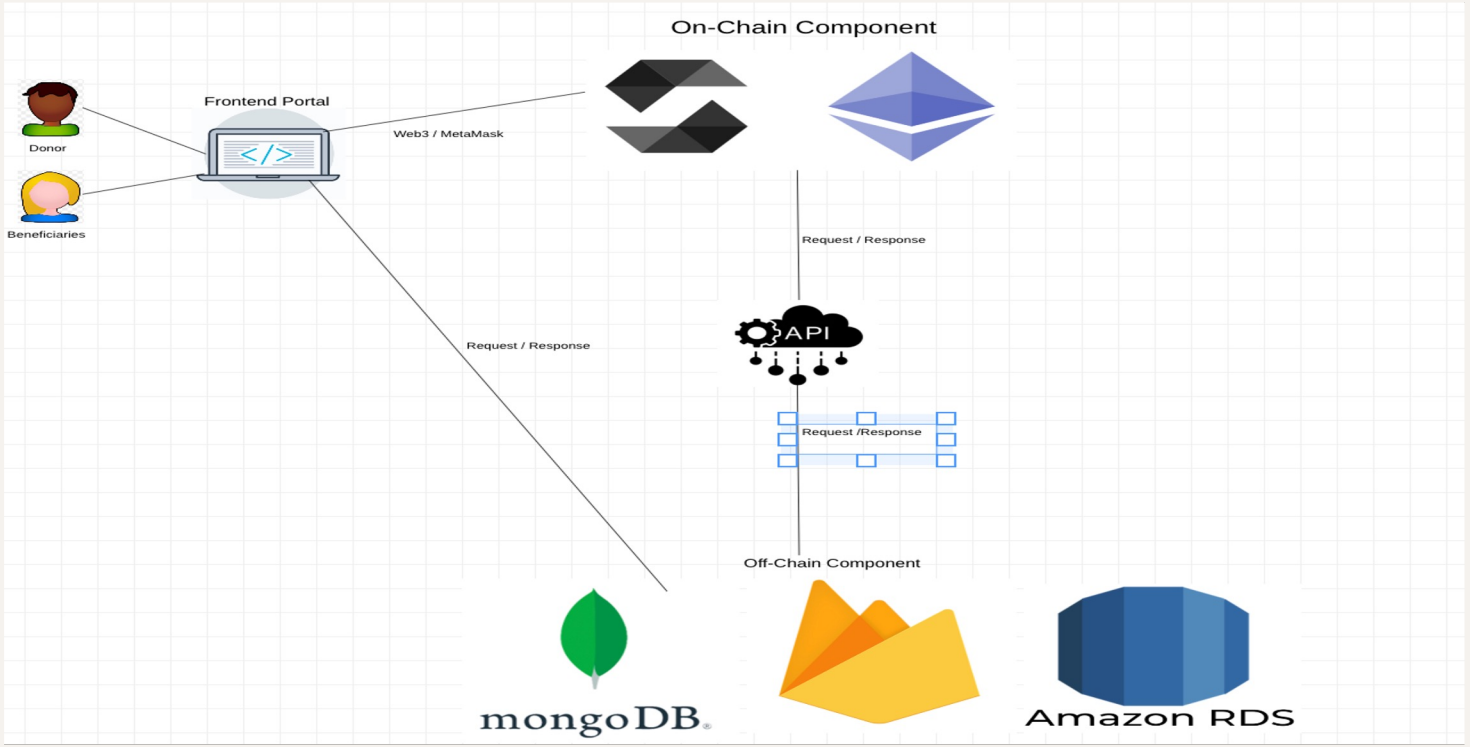


Flow Chart

Feedback loop:



Network Model



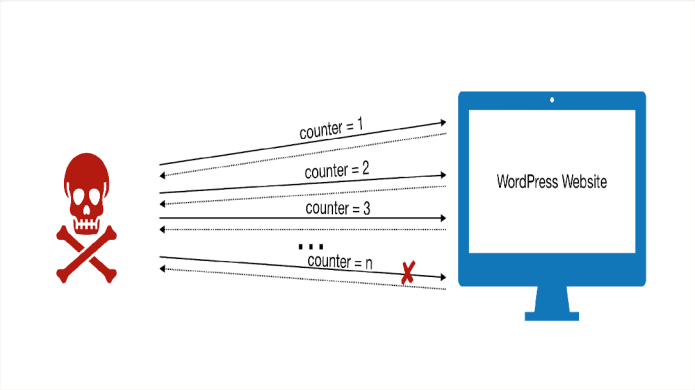
Functions

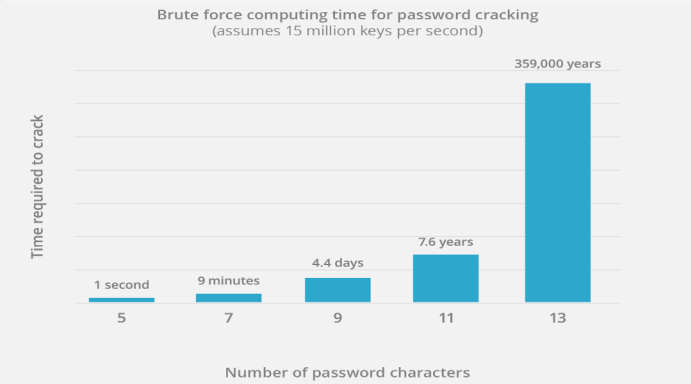
- Donators donate money or items to stores
- Beneficiaries request donations from stores
- Stores can receive donations from donators and distribute them to beneficiaries.
- Adding credit score for beneficiaries

Demonstration

Security & Threat Model

1. Brute force attack: Try to use all possible private keys to verify the signature or generate a fake signature

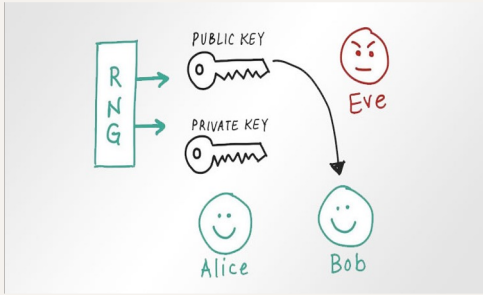




https://www.cloudflare.com/learning/bots/brute-force-attack/

Security & Threat Model

2. Random number attack: Crack the random number generation algorithm to predict the next random number.

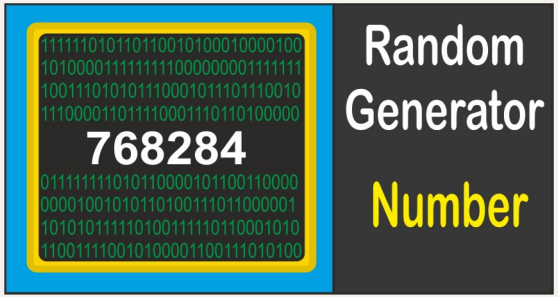


$$k \in (1 < k < q)$$

$$r = (G^k \pmod p) \mod q$$

$$s = (k^{-1} * m + xr) \mod q$$

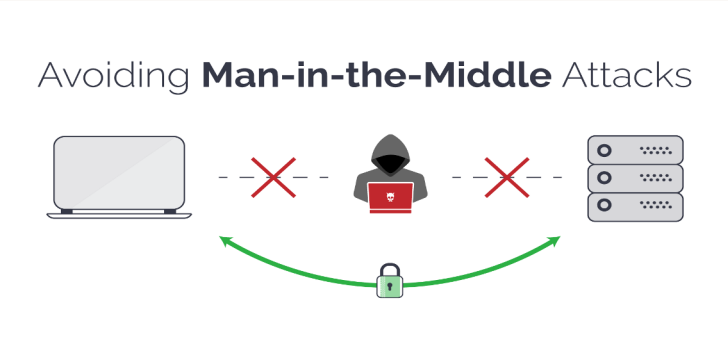
$$signature = (r, s)$$



HMAC_DRBG
HASH_DRBG

Security & Threat Model

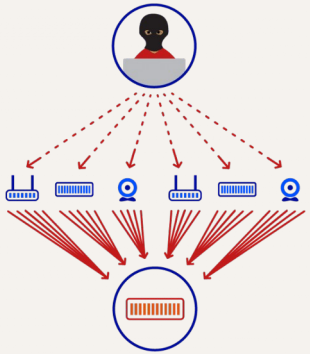
3. Man-in-the-middle attack: Tampering, forging public keys or signatures to deceive the verifier.



Digital certificates can be used to verify that both parties to a communication are legitimate and trustworthy

Security & Threat Model

4. Distributed Denial of Service attack: Overload the server or network resources by sending requests to the target server or network with a large amount of malicious traffic.



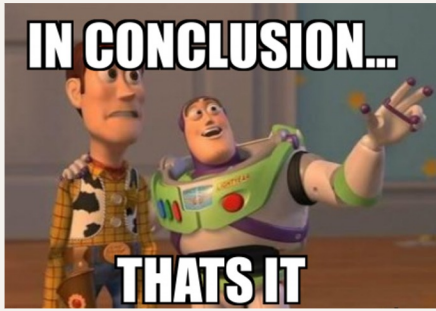
Restrict access to smart contracts to authenticated users or nodes

Security & Threat Model

5. 51% attack: The attacker controls more than 51% of the blockchain network computing power and is thus able to tamper with transaction records

PoA's authentication nodes consist of trusted entities, thus providing greater security and assurance

Conclusion



MileStone:

1. Design and implement a donation system that donates money or items to the store, requests donations from the store, and validates and responds to spending tokens
2. Complete smart contract components using digital signature algorithm.

Remaining Works:

1. Enhance security features to mitigate potential attacks, such as Man-in-the-middle attack
2. Conduct further experiments to validate users
3. Further develop front-end

Thank you