The 25th International Conference on Engineering of Complex Computer Systems (ICECCS2020)

Burn-to-Claim: a cross-blockchain asset transfer protocol

B. Pillai, K. Biswas, Z. Hou, V. Muthu

School of ICT, Griffith University, Gold Coast, Australia

March 5, 2021

Overview

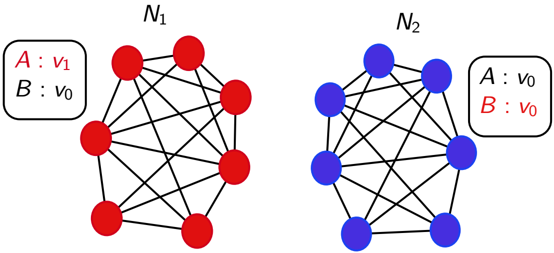
1. Blockchain - Interoperability & Challenges
2. Burn-to-Claim - Cross-Blockchain Asset Transfer Protocol
3. Theoretical Analysis - Security, Correctness & Fairness

Blockchain interoperability

- Information system interoperability refers to the ability to communicate and exchange information/data
- Blockchain it aims to **share** or **exchange** value or data
- But the design and architecture of the technology
  - limits to the transaction within the network
  - independent network has its own state assumptions and
  - one network can not verify the state of another network
- Therefore, interoperability is challenging to reach consensus

Interoperability challenges

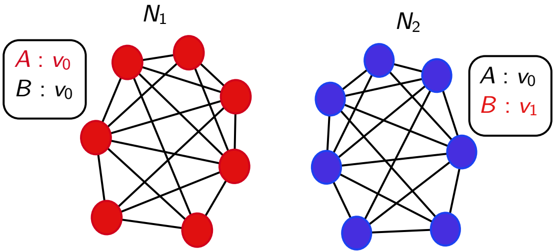
- Technical difficulty to trust and accept external data



- Assumptions
  - N_1 and N_2 are secure
  - v represent an asset-token
  - merge mining and gateway node

Interoperability challenges

- Technical difficulty to trust and accept external data



- Cross-blockchain properties
  1. Security
  2. Correctness
  3. Fairness

Burn-address

Public key (K_p) and Private key (K_r)

genAddrress(K_p) $\rightarrow$ blockchain address (K_{adr})

Tx[Sender $\rightarrow$ Recipient : value]$_{sign}$ $Tx[K^S_{adr} \rightarrow K^R_{adr}: v]K^S_r$

Definition (Burn-address)

A burn-address given as β is an address to which one can send assets, but they can never be recovered because the private key of the corresponding address is not known/ accessible.

$Tx[K^S_{adr} \rightarrow \beta: v]K^S_r$

Karantias et. al., (2020). **Proof-of-burn**. In International Conference on Financial Cryptography and Data Security

Burn-address

Public key (K_p) and Private key (K_r)

genAddrress(K_p) $\rightarrow$ blockchain address (K_{adr})

Tx[Sender $\rightarrow$ Recipient : value]$_{sign}$ $\quad Tx[K^S_{adr} \rightarrow K^R_{adr}: v]K^S_r$

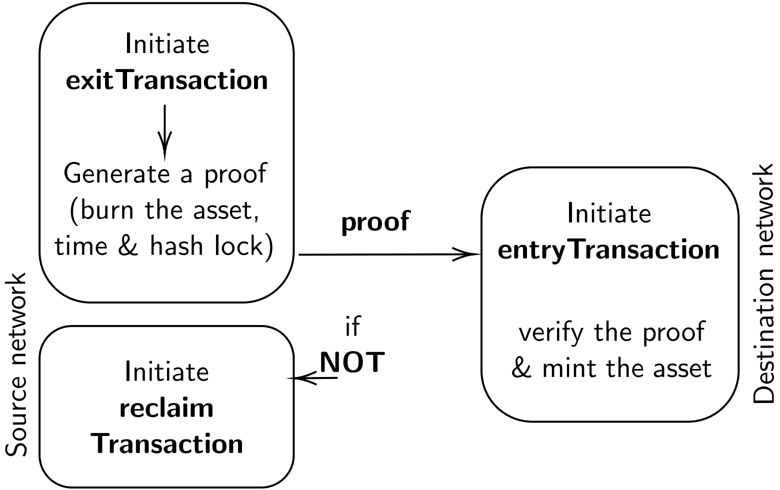
Definition (Burn-address)

A burn-address given as β is an address to which one can send assets, but they can never be recovered because the private key of the corresponding address is not known/ accessible.

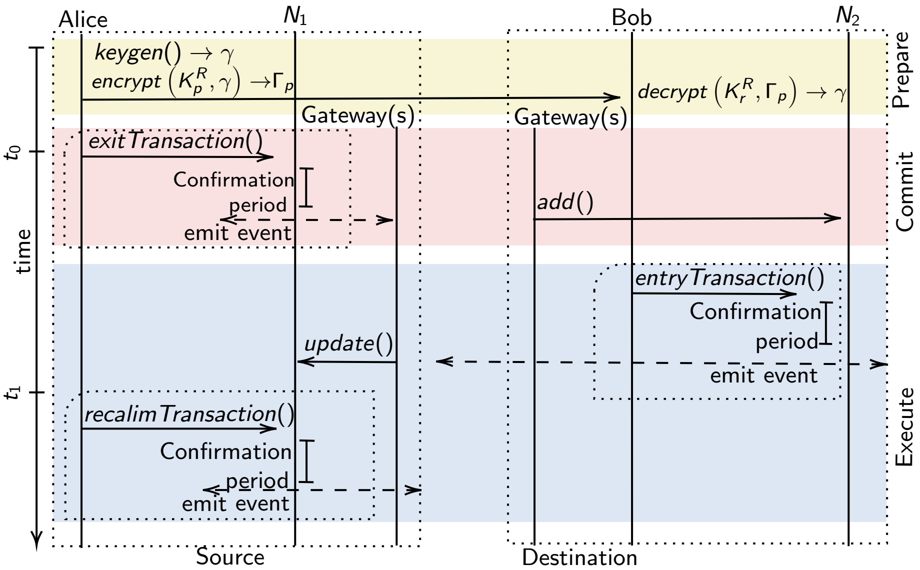
$Tx[K^S_{adr} \rightarrow \beta: v]K^S_r \qquad Tx[\beta \rightarrow K_{adr}: v]K_r$

Karantias et. al., (2020). **Proof-of-burn**. In International Conference on Financial Cryptography and Data Security

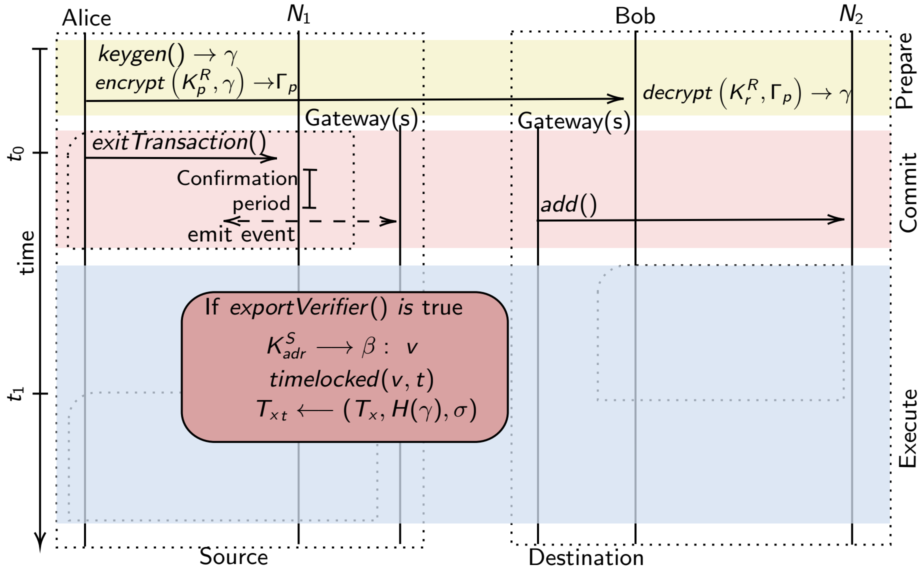
Proposed Burn-to-Claim protocol



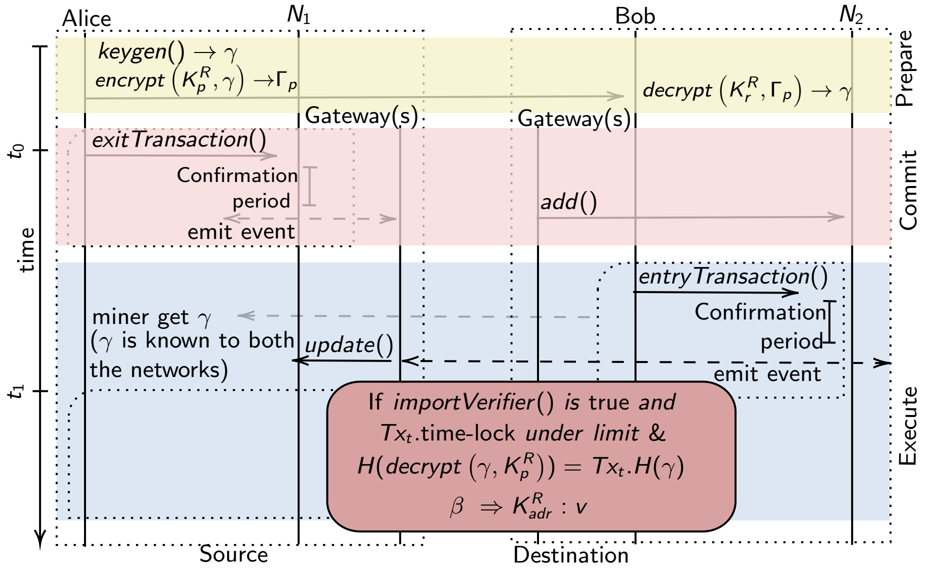
Protocol overview



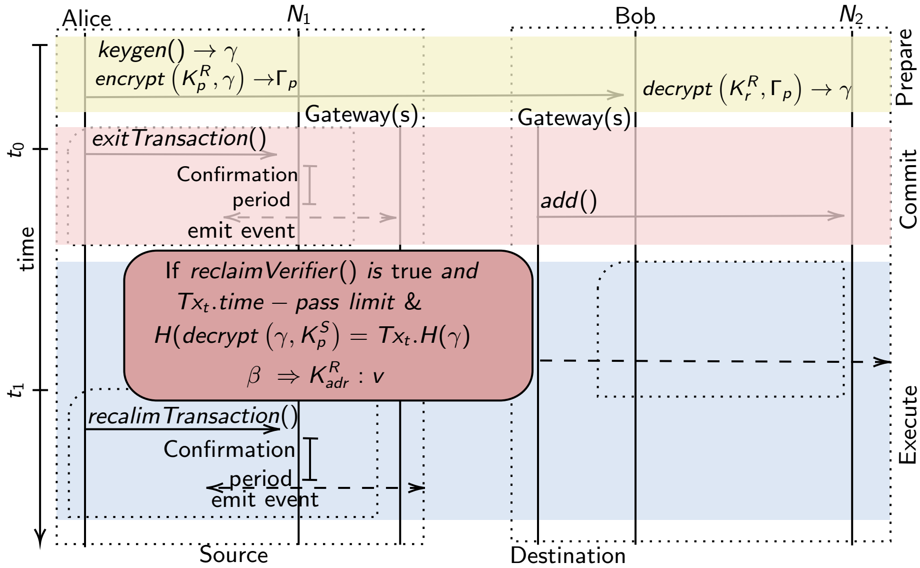
Protocol overview - commit stage



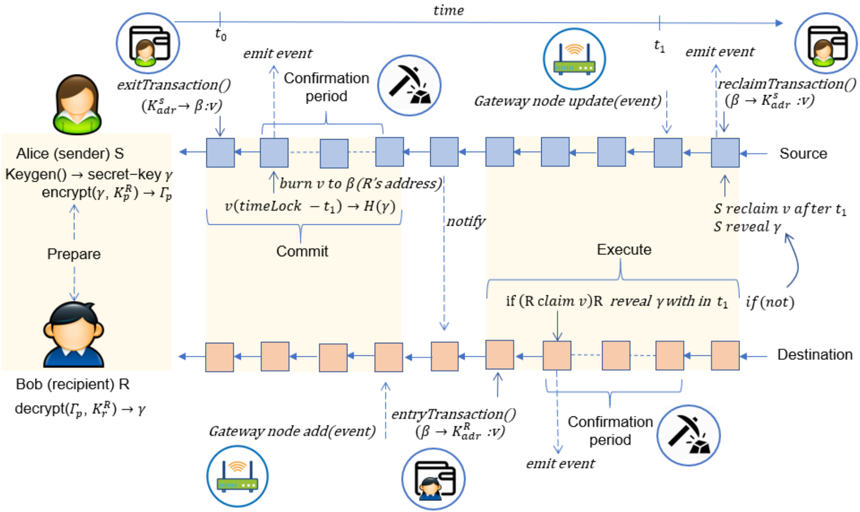
Protocol overview - execute stage



Protocol overview - execute stage



Burn-to-Claim protocol case study



Theoretical analysis - 1) security

The recipient network can rely on the Burn-to-Claim exit-proof guarantee provided by the source network

- Burn-address β is unspendable with respect to a proof-of-burn protocol
- exitTransaction function transfer asset v to a β, which is unspendable

Theoretical analysis - 2) correctness

The exchange operation only exchange an asset to a correct recipient

- β is derived from the recipient's address K_{adr}^{R}
- Recipient must prove the ownership of address
- Nodes verify signature
- Only the user who owns a private key associated with K_{adr}^{R} can make the claim for the asset in the destination network

Theoretical analysis - 3) fairness

The exchange operation should only yield one of the following outcomes; either the exchange succeeds and the asset is transferred to the recipient; or it failed and the asset must return to the sender

- For recipient to claim v, recipient must reveal γ there after S can not claim it
  - because γ is known to both the networks
- If R fails to claim, after the time-lock period S can re-claim v
- With the hash-time-lock mechanism the transfer can be guaranteed atomicity, without a trusted third party, thus the protocol satisfies the fairness property

Future work

- Implement and test the protocol for an application environment
- Time and cost analysis
  - *time* - transfer time from N_1 to N_2
  - *cost* - gas cost to perform the transfer
- Develop a threat model - selfish mining and gateways
  - evaluating using formal analysis

thank you...

babu.pillai@griffithuni.edu.au