

Provisions

Privacy-preserving proofs of solvency for Bitcoin exchanges

Real World Crypto 2016

eprint.iacr.org/2015/1008.pdf

github.com/bbuenz/provisions

Gaby Dagher



Benedikt Bünz



Joseph Bonneau



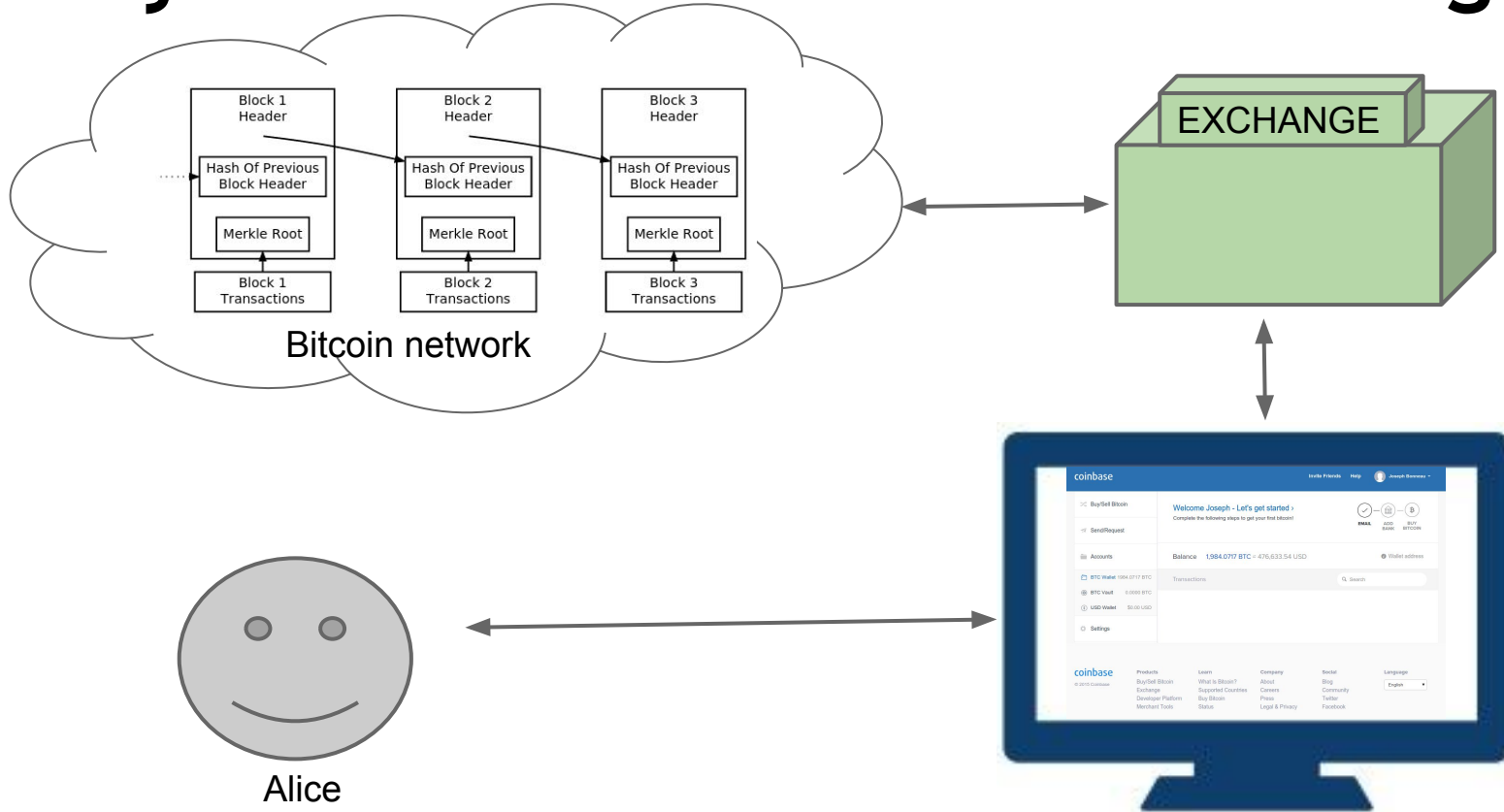
Jeremy Clark



Dan Boneh



Many users use Bitcoin via *exchanges*



Exchanges look a lot like online banks

coinbase

Invite Friends

Help



Joseph Bonneau ▾

Buy/Sell Bitcoin

Welcome Joseph - Let's get started >

Complete the following steps to get your first bitcoin!



EMAIL



ADD
BANK



BUY
BITCOIN

Send/Request

Accounts

Balance 1,984.0717 BTC ≈ 476,633.54 USD

Wallet address

BTC Wallet 1984.0717 BTC

Transactions

Search

BTC Vault 0.0000 BTC

Exchanges have a shaky track record



Mt. Gox:
lost roughly US\$450M
Subsequent price crash

~50% have failed!
[Moore, Christin 2013]

Goal: prove *solvency*



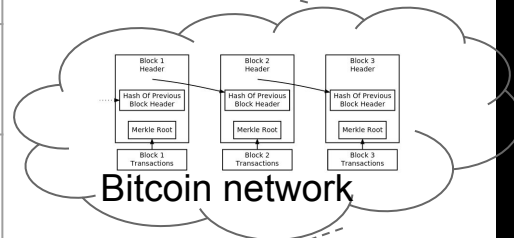
USERS

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$



BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$



Solvency \Leftrightarrow Total Liabilities \leq Total Assets

full reserve

Proofs of solvency have limitations

- Proof of solvency is a snapshot
- Proof of solvency \neq willingness to pay

Approach #1: publish everything



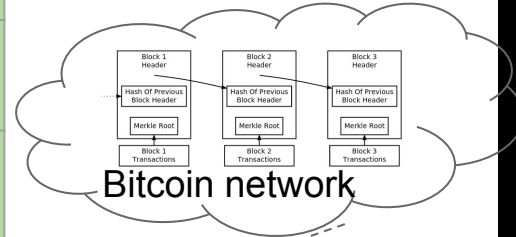
USERS

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$



BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$



Approach #2: trusted auditor



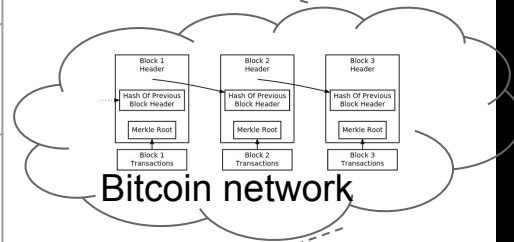
USERS

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$



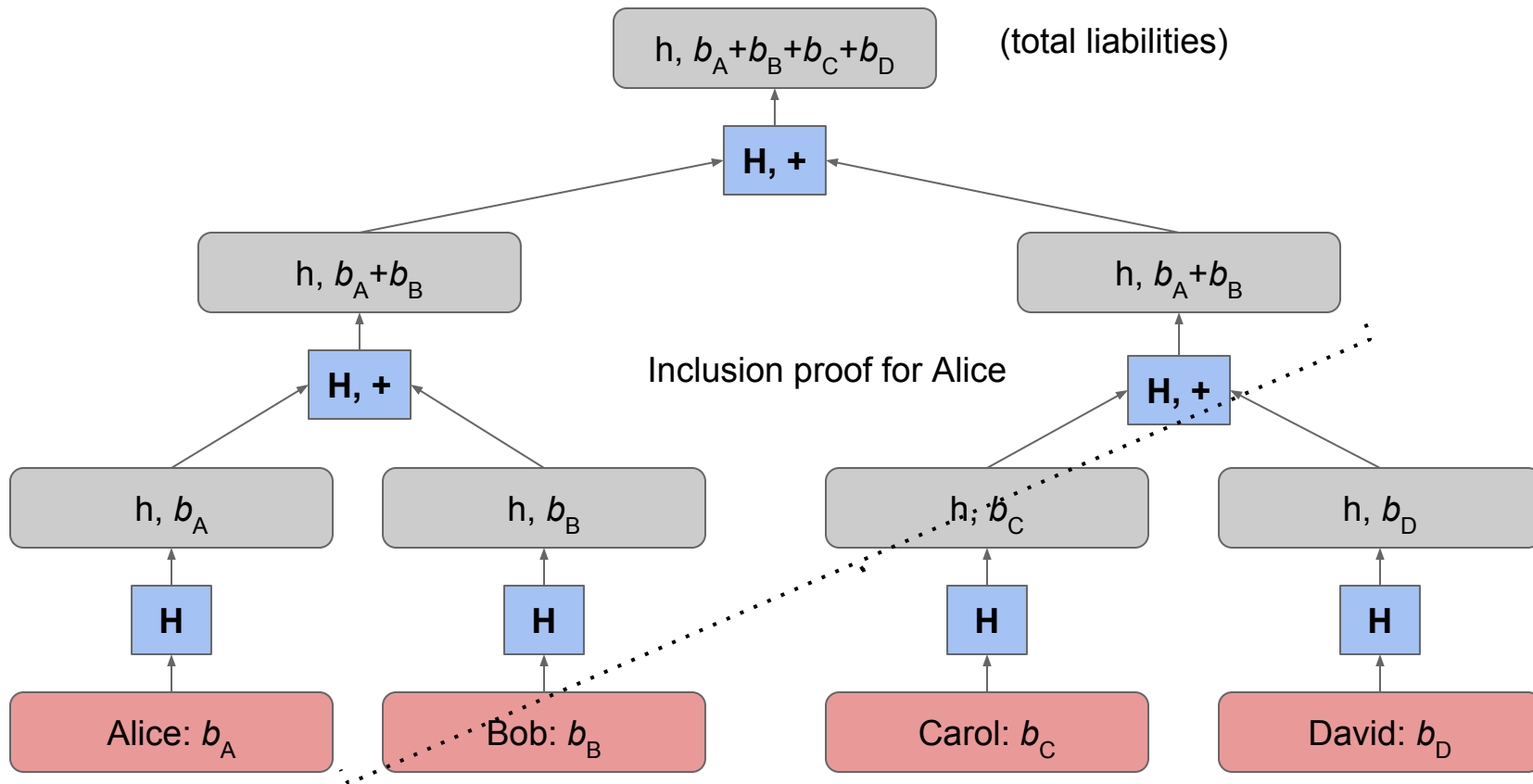
BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$



Looks good to me!

Solution #3a: Maxwell protocol [2013]



Solution #3b: public proof of assets



The image is a screenshot of the CoinDesk website. At the top left is the CoinDesk logo. To its right, under the heading "TRENDING", is a link to "CoinDesk's Report on Banking and the Blockchain Now Available". On the top right, there is a "BITCOIN PRICE INDEX (24H)" section with a line chart showing price fluctuations between \$429 and \$443. Below the header is a yellow navigation bar with links for NEWS, PRICE & DATA, GUIDES, EVENTS, RESEARCH, and PRESS RELEASES. The main content area shows a breadcrumb trail "BITSTAMP • COMPANIES • NEWS" and a large article title: "Bitstamp Audit Proves it was Behind \$147 Million Mystery Bitcoin Wallet". Below the title is the author "Pete Rizzo (@pete_rizzo_)" and the publication date "Published on March 6, 2014 at 17:41 GMT".

CoinDesk

TRENDING
CoinDesk's Report on Banking and the Blockchain Now Available

BITCOIN PRICE INDEX (24H)
\$443
\$436
\$429

NEWS ▾ PRICE & DATA ▾ GUIDES ▾ EVENTS ▾ RESEARCH ▾ PRESS RELEASES ▾

BITSTAMP • COMPANIES • NEWS

Bitstamp Audit Proves it was Behind \$147 Million Mystery Bitcoin Wallet

Pete Rizzo (@pete_rizzo_) | Published on March 6, 2014 at 17:41 GMT

Maxwell protocol considered too leaky



“Maxwell’s proposal would have required bitcoin companies to reveal all of their balance-containing addresses. This method would result in the public knowledge of exchanges’ or wallet providers’ bitcoin wallets and total holdings, information that is commercially sensitive and presents potential security risks to companies and users.”

Improving on Maxwell's privacy goals

Maxwell protocol reveals:

- Total liabilities
- Some info about account sizes
- Total assets
- Addresses in use

Non-goal: completely conceal number of users

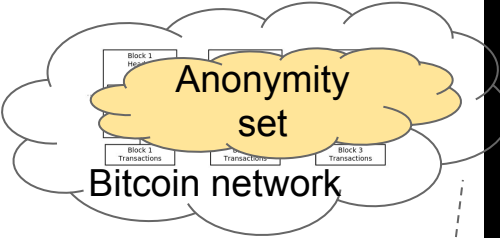
Provisions at a high level

 USERS

 BITCOIN ADDRESSES

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$



Proof-of-liabilities

Proof-of-assets

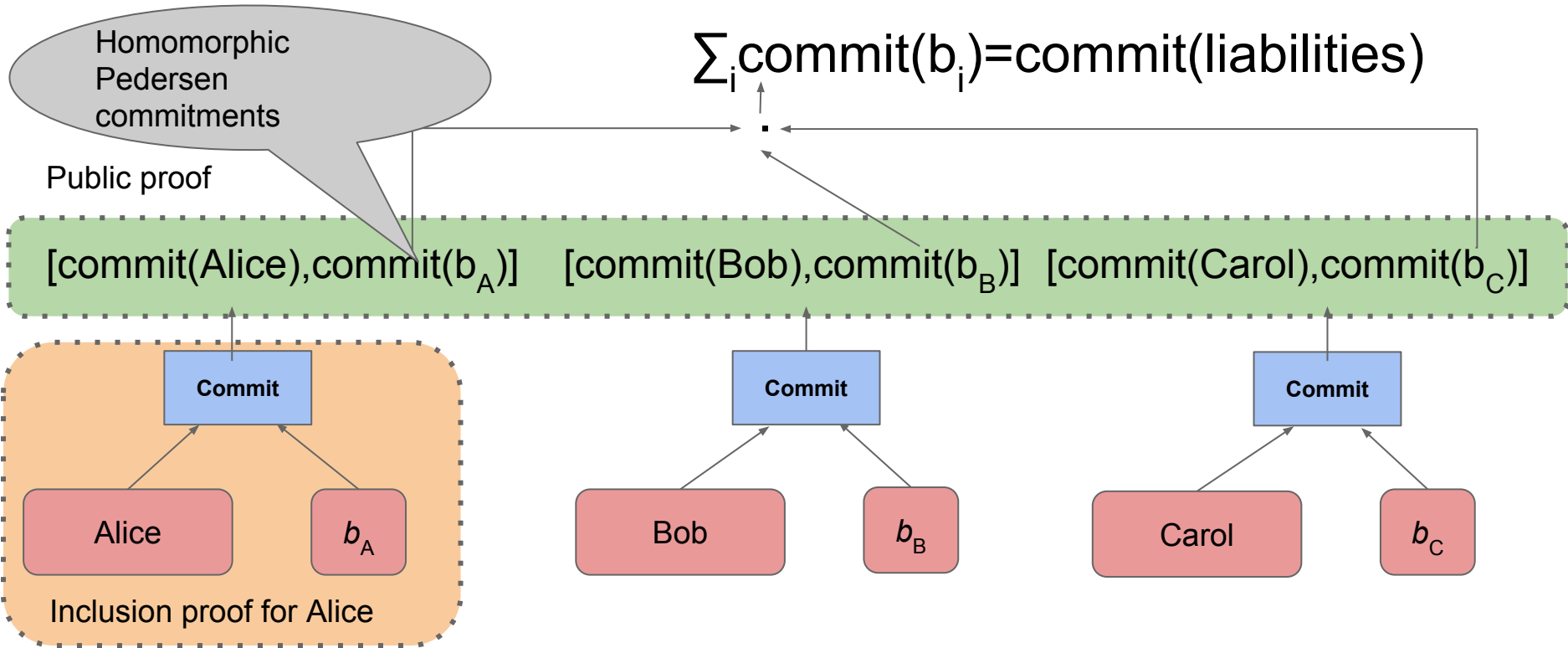
commit(liabilities)

commit(assets)

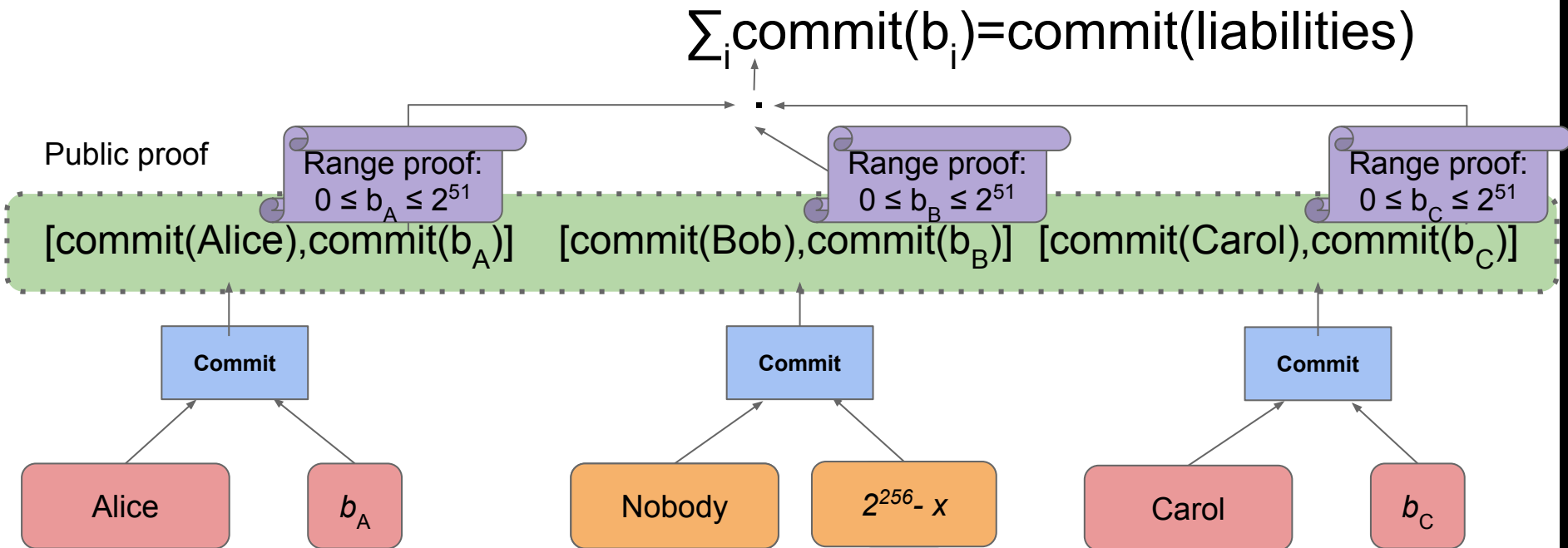
Proof-of-solvency

commit(assets - liabilities) = commit(0)

Provisions proof-of-liabilities



Provisions proof-of-liabilities



What if a fake user causes an overflow?

\Rightarrow *range proof* needed for each committed balance

Size of proof-of-liabilities

- Proof size is $\Theta(m \cdot n)$ for n users, m bits precision
- ~9kB/user at 51 bits (31 bits should be enough)
- easily parallelizable
- incrementally updatable

Provisions at a high level



USERS

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$

Proof-of-liabilities

commit(liabilities)

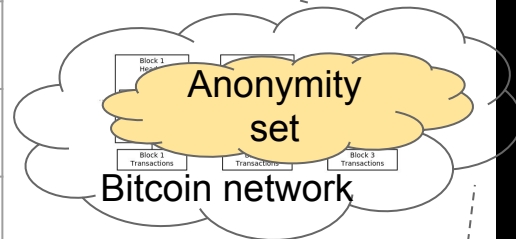


BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$

Proof-of-assets

commit(assets)



Proof-of-solvency

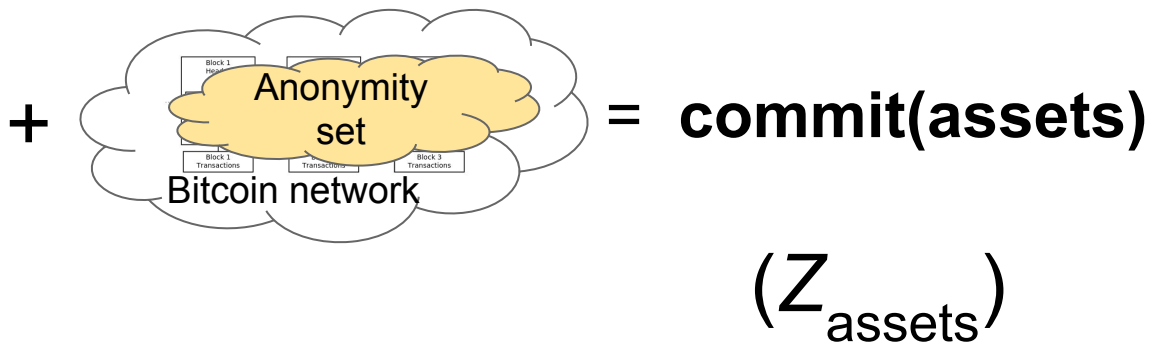
commit(assets - liabilities) = commit(0)

Provisions proof-of-assets



BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$

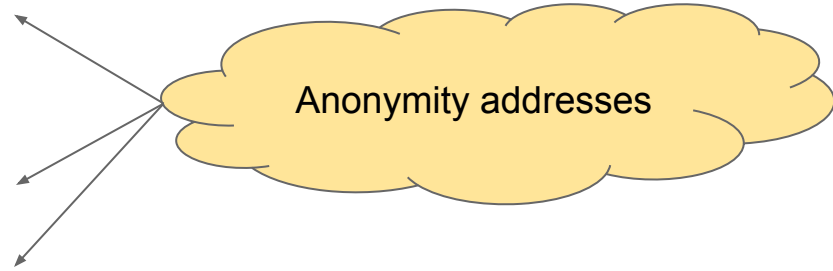


NIZKPK:

-exchange knows private keys for a subset of Bitcoin addresses -
total value at these addresses is committed to by Z_{assets}

Provisions proof-of-assets


private key	address	<i>public balance</i>
k_1	K_1	b_1
?	K_2	b_2
k_3	K_3	b_3
?	K_4	b_4
?	K_5	b_5
k_6	K_6	b_6



Provisions proof-of-assets

private key	address	<i>public balance</i>	<i>committed balance</i>
k_1	K_1	b_1	commit(b_1)
?	K_2	b_2	commit(0)
k_3	K_3	b_3	commit(b_3)
?	K_4	b_4	commit(0)
?	K_5	b_5	commit(0)
k_6	K_6	b_6	commit(b_6)

commitments to 0



Provisions proof-of-assets

Public proof

private key	address	public balance	committed balance	per-address proof
k_1	K_1	b_1	$p_1 = \text{commit}(b_1)$...
?	K_2	b_2	$p_2 = \text{commit}(0)$...
k_3	K_3	b_3	$p_3 = \text{commit}(b_3)$...
?	K_4	b_4	$p_4 = \text{commit}(0)$...
k_6	K_6	b_6	$p_6 = \text{commit}(b_6)$...

“Either I know k_i and p_i is a commitment to b_i
OR p_i is a commitment to 0”

$$\sum_i p_i = \text{commit}(\text{assets})$$

Size of proof-of-assets

- Proof size is $\Theta(N)$ for N addresses in anonymity set
- ~350 bytes/address
 - 1 public key
 - 2 elements of G ,
 - 8 elements of \mathbf{Z}_q
- easily parallelizable

Completing the proof of solvency

 USERS

Alice	b_A
Bob	b_B
Charlie	b_C
...	
TOTAL LIABILITIES	$b_A + b_b + b_c + \dots$

Proof-of-liabilities

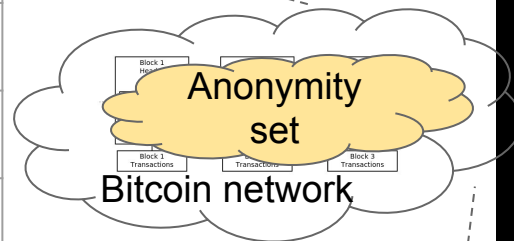
commit(liabilities)

 BITCOIN ADDRESSES

K_1	b_1
K_2	b_2
K_3	b_3
...	
TOTAL ASSETS	$b_1 + b_2 + b_3 + \dots$

Proof-of-assets

commit(assets)



Proof-of-solvency

commit(balance) = commit(assets) - commit(liabilities)

Finishing the proof of solvency in style

Given:

$$\text{commit}(\text{balance}) = \text{commit}(\text{assets}) - \text{commit}(\text{liabilities})$$

- open $\text{commit}(\text{balance})$ \Rightarrow reveals surplus
- range proof that $\text{commit}(\text{balance})$ is small \Rightarrow proof that *surplus* exists

Extension: Valet keys

Keys are stored offline



Extension:

- replace \mathbf{g}^x for every key with \mathbf{g}^{xr}
- Prove knowledge of each \mathbf{g}^{xr} to the base \mathbf{g}^x
- xr is the **valet key, safe to export**

Provisions is practical

- 150 MB asset proof with maximal anonymity set
- 17 GB proof of liabilities for 2 Million users (Coinbase)
- Computes in ~ 1 hour on 1 machine
- Auditors check entire proof (~ 1 hour)
- Users verify inclusion (~ free)

Limitation: non-public public keys

- Provisions requires public keys for entire anonymity set
- Most bitcoin addresses are $H(\text{PubKey})$
 - Public key revealed after first spend
 - Majority are one-time use...
- About 430k/1.3M addresses can be used in Provisions

⇒ *SNARKs* could be used to build a more powerful solvency proof.

Thanks!

buenz@cs.stanford.edu

Paper:

eprint.iacr.org/2015/1008.pdf

Reference implementation:

github.com/bbuenz/provisions