TÜBİTAK
BİLGEM
UEKAE

BLOKZİNCİR
Araştırma Laboratuvarı

Zero-Knowledge Proofs in Blockchains (Blokzincirlerde Sıfır Bilgi İspatları)

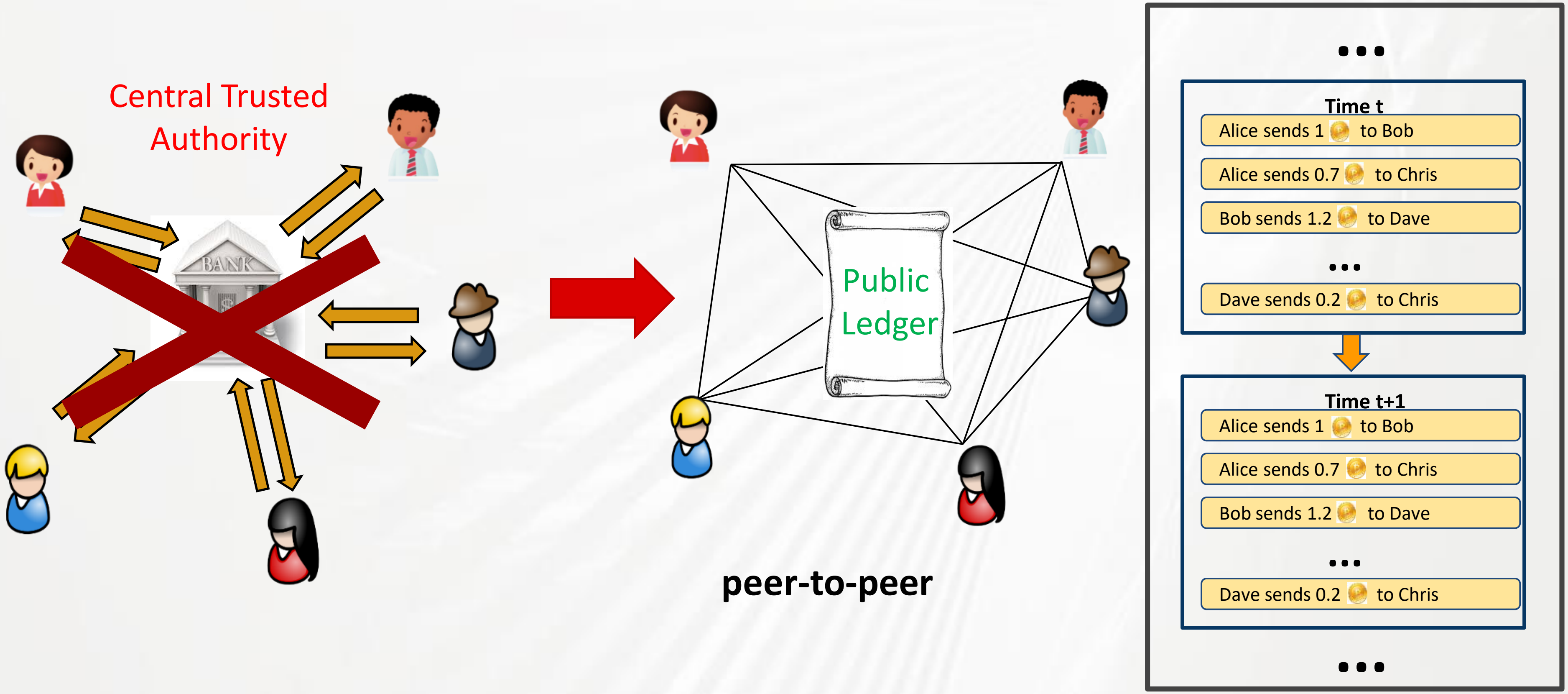
2. Uusal Blokzincir Çalıştayı, İstanbul, 2019
26 Eylül 2019

Dr. Muhammed Ali BİNGÖL

muhammedali.bingol@tubitak.gov.tr

Privacy against who?

❑ Decentralized, no trusted server



…

Time t

Alice sends 1 ● to Bob

Alice sends 0.7 ● to Chris

Bob sends 1.2 ● to Dave

…

Dave sends 0.2 ● to Chris

Time t+1

Alice sends 1 ● to Bob

Alice sends 0.7 ● to Chris

Bob sends 1.2 ● to Dave

…

Dave sends 0.2 ● to Chris

…

Centralized: Reveal amount, sender/receiver info to the bank

De-centralized: Reveal amount, sender/receiver info to everyone

Transaction amounts available in the clear

Everyone can see the payer, payee, and value

Business implications:

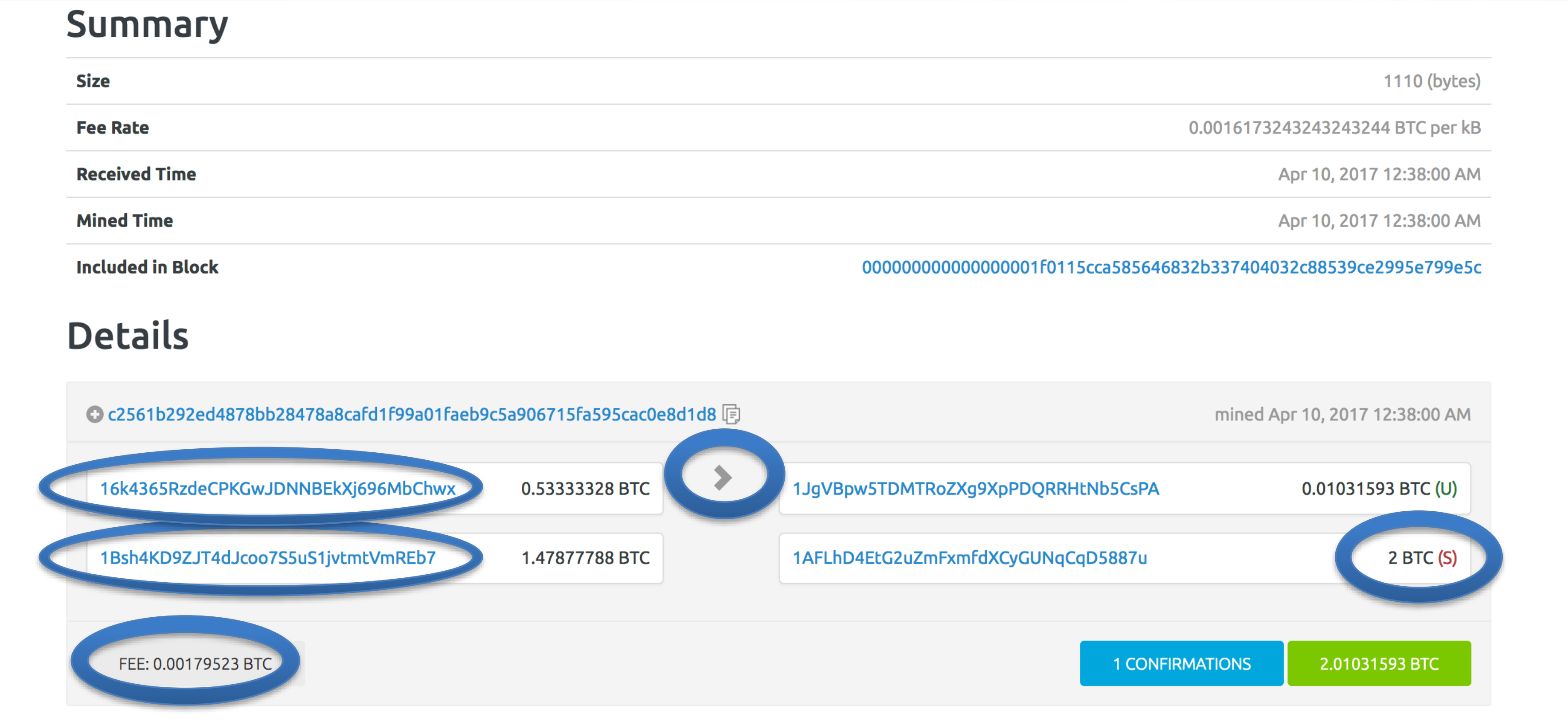
- Company pays employees in Bitcoin.
  ⇒ all salaries are public

- Public supply chain prices:
  - How much does Ford pay its supplier for tires?

Problem: Every transaction ever made is **recorded forever**

Bitcoin is neither confidential nor anonymous

Anonymity vs Pseudnymity



Summary

Size	1110 (bytes)
Fee Rate	0.0016173243243243244 BTC per kB
Received Time	Apr 10, 2017 12:38:00 AM
Mined Time	Apr 10, 2017 12:38:00 AM
Included in Block	000000000000000001f0115cca585646832b337404032c88539ce2995e799e5c

Details

c2561b292ed4878bb28478a8cafd1f99a01faeb9c5a906715fa595cac0e8d1d8 — mined Apr 10, 2017 12:38:00 AM

16k4365RzdeCPKGwJDNNBEkXj696MbChwx	0.53333328 BTC	>	1JgVBpw5TDMTRoZXg9XpPDQRRHtNb5CsPA	0.01031593 BTC (U)
1Bsh4KD9ZJT4dJcoo7S5uS1jvtmtVmREb7	1.47877788 BTC		1AFLhD4EtG2uZmFxmfdXCyGUNqCqD5887u	2 BTC (S)

FEE: 0.00179523 BTC

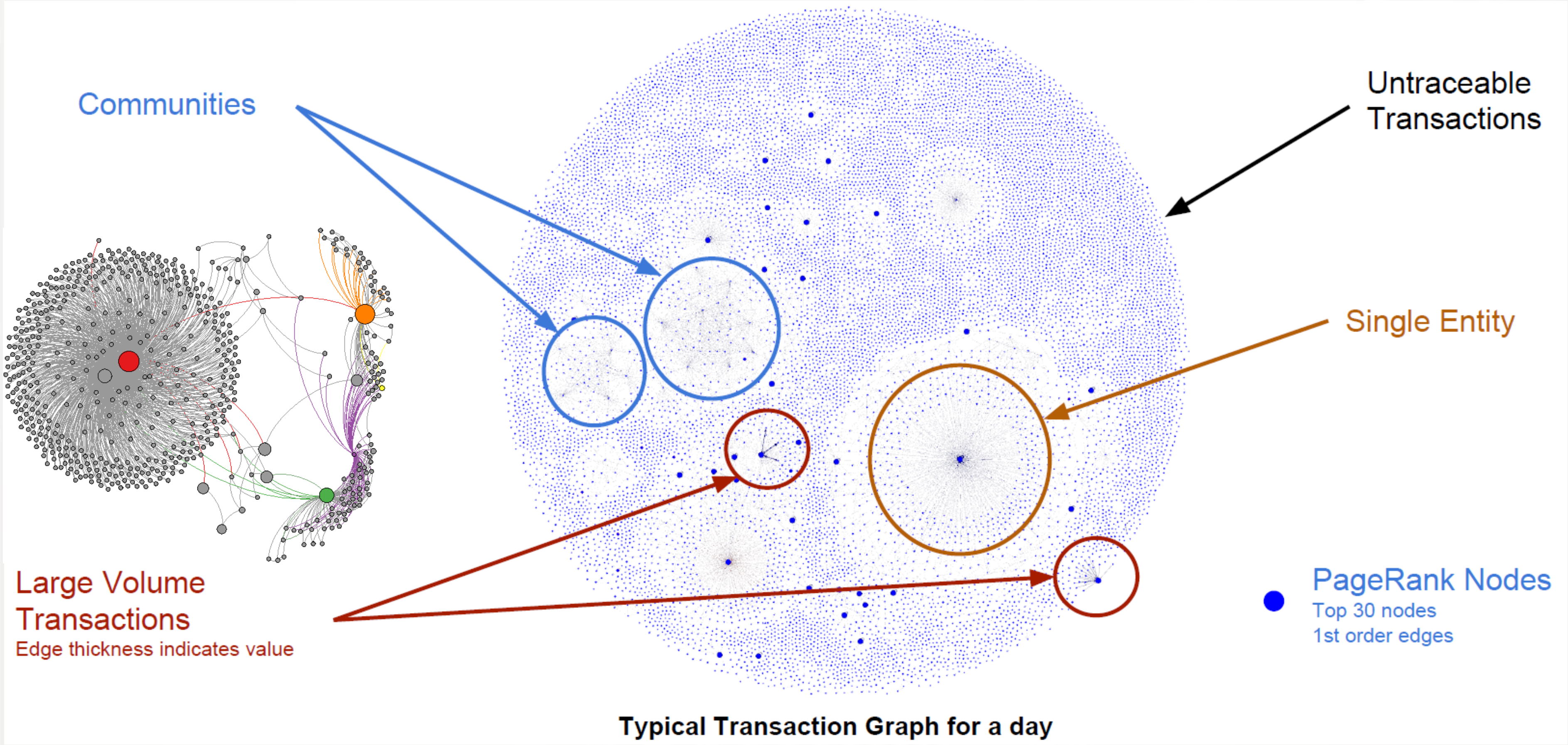
1 CONFIRMATIONS

2.01031593 BTC

Bitcoin only offers pseudo-anonymity. Transactions are linkable and can be potentially de-anonymized

Transaction Flow Graph [Maxwell 2016]

Pseudonymity cannot provide Anonymity!!

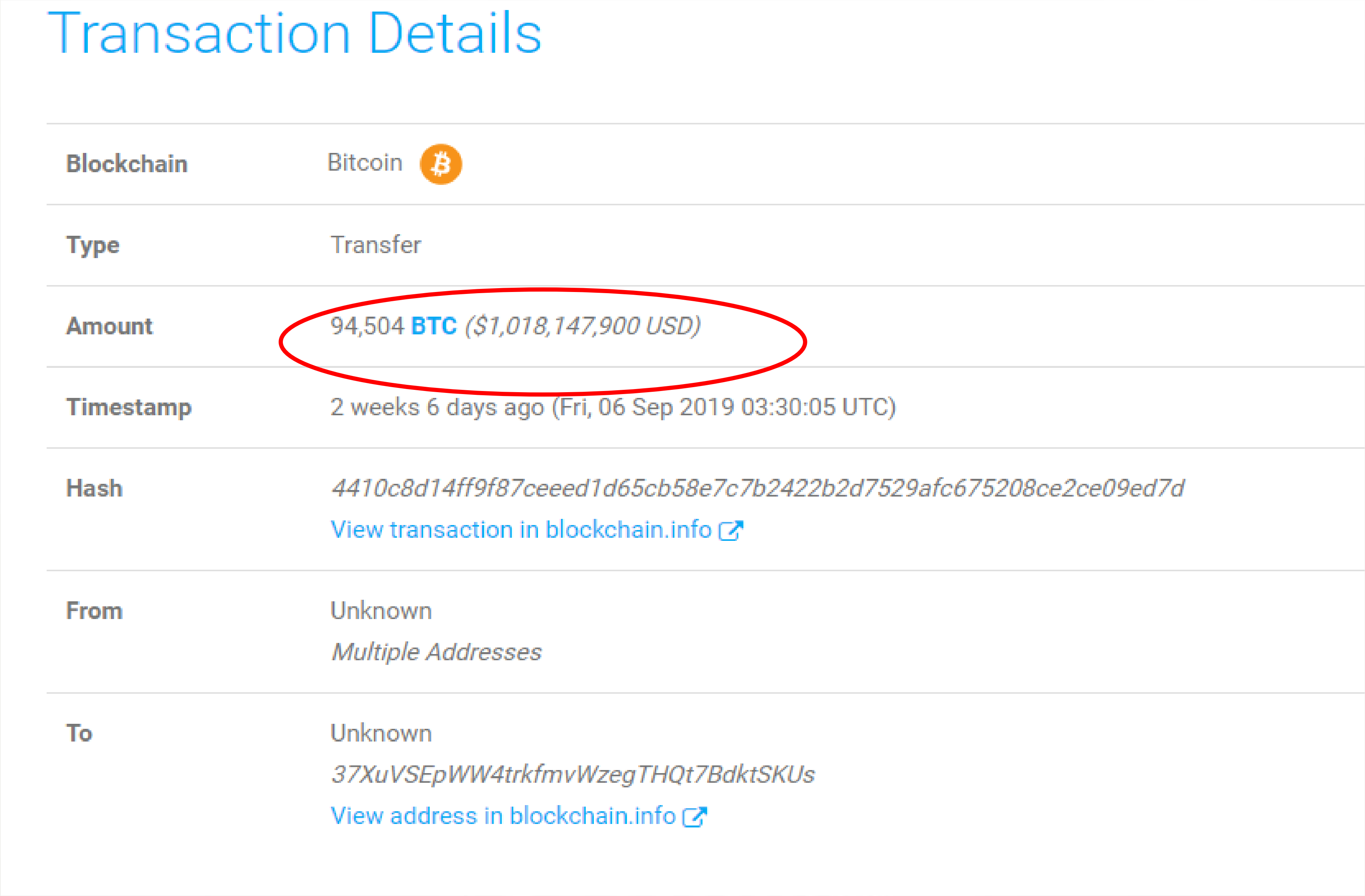


Typical Transaction Graph for a day

- **Transaction graph is still public**

[Reid Martin 11] [Barber Boyen Shi Uzun 12] [Ron Shamir 12] [Ron Shamir 13] [Meiklejohn Pomarole Jordan Levchenko McCoy Voelker Savage 13] [Ron Shamir 14]

Breaking News on Transactions

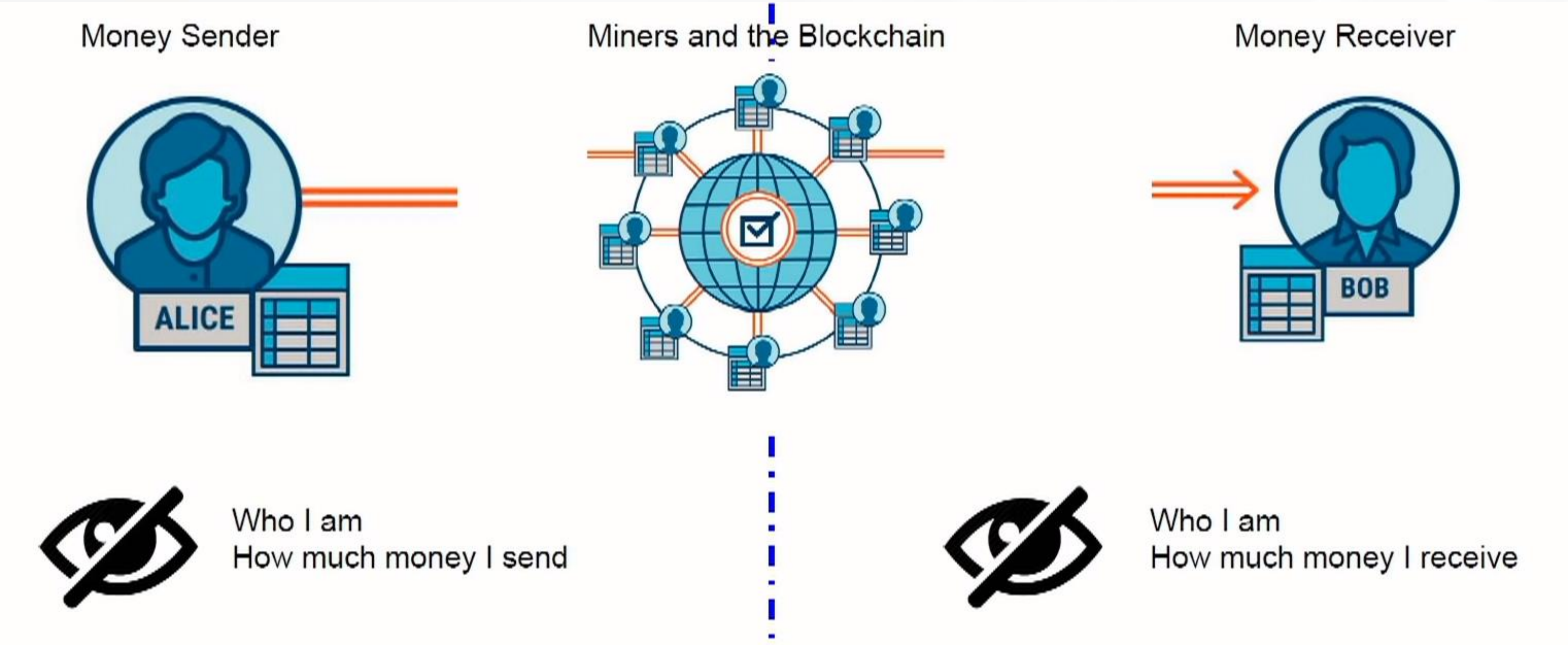


Transaction Details

Blockchain	Bitcoin ₿
Type	Transfer
Amount	94,504 **BTC** *($1,018,147,900 USD)*
Timestamp	2 weeks 6 days ago (Fri, 06 Sep 2019 03:30:05 UTC)
Hash	*4410c8d14ff9f87ceeed1d65cb58e7c7b2422b2d7529afc675208ce2ce09ed7d* View transaction in blockchain.info
From	Unknown *Multiple Addresses*
To	Unknown *37XuVSEpWW4trkfmvWzegTHQt7BdktSKUs* View address in blockchain.info

Transactions are public

Confidentiality and Anonymity



1 **Confidentiality**

hiding the transferred **amounts**

2 **Sender Anonymity**

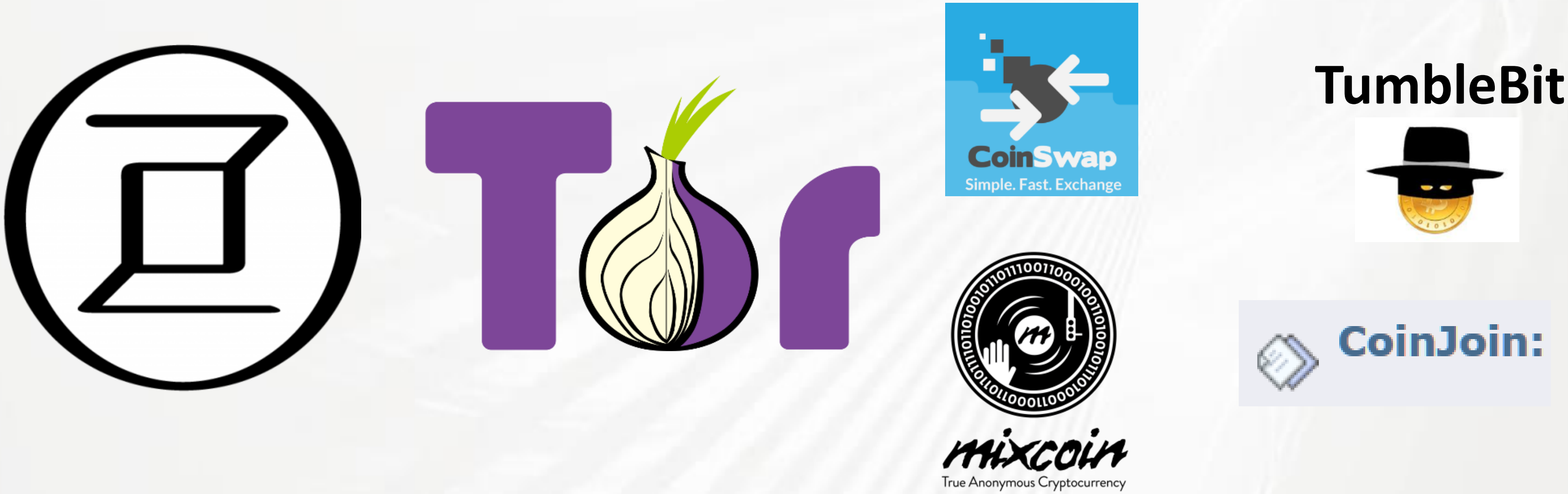
hiding the identities of the sender / the transaction origins

3 **Receiver Anonymity**

hiding the recipients identity

Bitcoin is not anonymous....what is next?

Option 1: minting/burning, mixers/tumblers compatible with Bitcoin



Option 2: New coin based on Zero Knowledge proofs

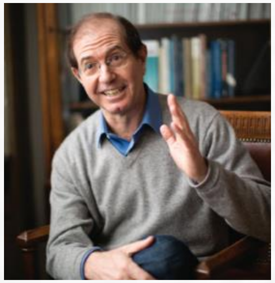
...

Zero-Knowledge Proofs

Sıfır Bilgi Ispatları

Introducing Zero-Knowledge Protocols [GMR'85]

Zero-Knowledge Proofs [Goldwasser-Micali-Rackoff'85]

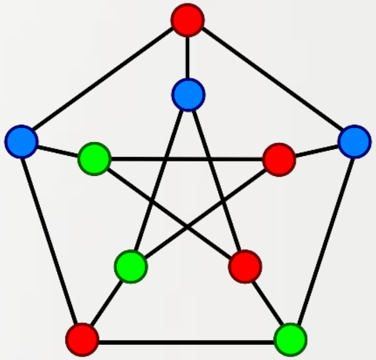


Every statement that has a classical proof (in **NP**) has zero-knowledge interactive proof, if **one-way functions** exists.

[Goldreich-Micali-Wigderson'91]

- There exists a ZK proof system for the NP-complete graph colouring problem with three colours.

[1] Goldreich, Oded; Micali, Silvio; Wigderson, Avi (1991). "Proofs that yield nothing but their validity". *Journal of the ACM*. **38** (3): 690–728.



http://web.mit.edu/~ezyang/Public/graph/svg.html

Secure Computation vs. Communication

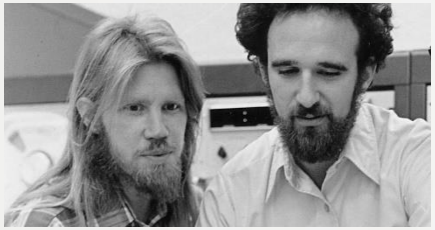
□ Secure Communication

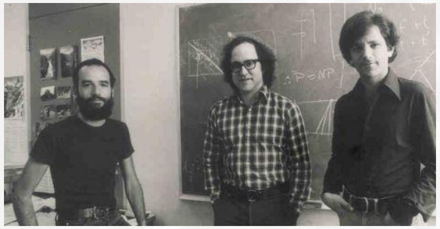
- Symetric-Key Cryptography
  - Block Ciphers
  - Stream Ciphers
  - Hash Functions
- Public-Key Cryptography
  - Asymmetric Encryption
  - Signature Schemes
- Access Control
- Etc.

≠

□ Secure Computation

- Secure Multi-party Computation
- Zero-Knowledge Protocols
  - Fiat-Shamir Protocol
  - Schnorr Proofs
  - Zk-Snarks
  - Zk-Starks
  - Bulletproofs
  - Sigma Bulletproofs etc.
- Private Function Evaluation
- Homomorphic Schemes
- Etc.





Can we achieve full privacy with ZKPs ?

ZKPs ≠ privacy

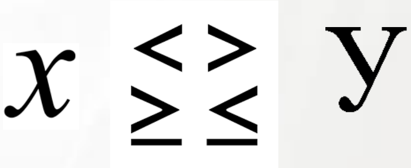
ZKPs == honest computation

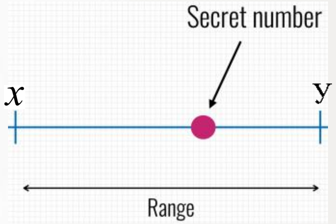
$$f(x) = y$$

+ proof

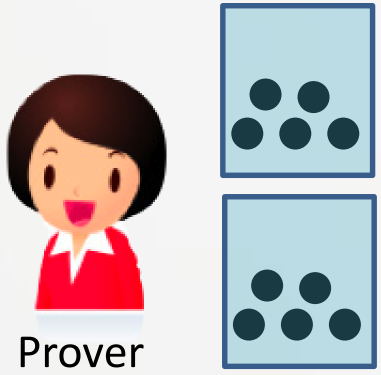
What you can prove using zero-knowledge proofs?

- There are four common statement types, though the following is not an exhaustive list:
  - • An **equality** statement (the subject’s bank account balance is equal to x), or **non-equality** statement.
  - • An **inequality** statement (the subject’s bank account balance exceeds x).
  - • A **range** statement (the subject’s bank account balance is within interval [a,b]), or out-of-range statement.
  - • A **membership** statement (the subject is on the client list of bank X), or non-membership statement.





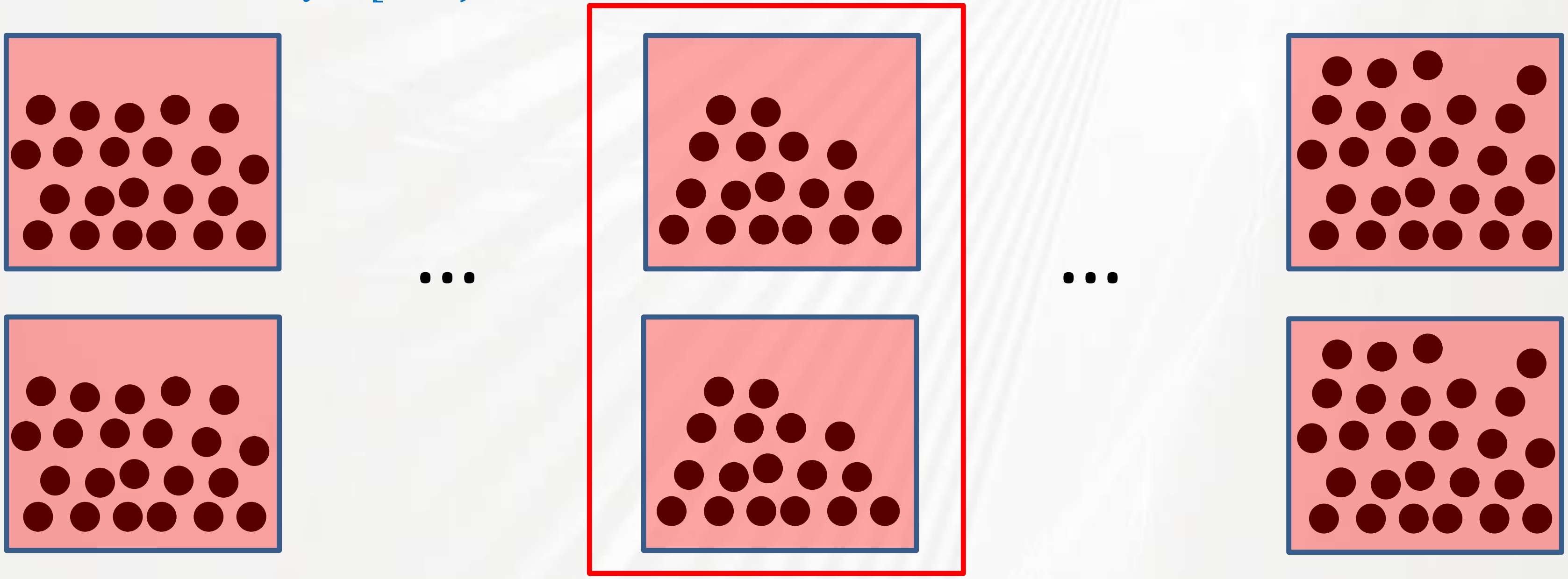
Zero-Knowledge Protocols – Equality Proof Example



Alice has two cups each containing $x \in [0, n)$ marbles.

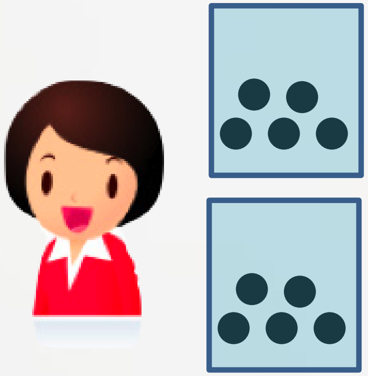
She wants to prove to Bob that both contain the same number without revealing x.

Alice prepares 10 pairs of buckets, both buckets in the i^{th} pair containing a random number $R_i \in [0, N)$ of marbles.



Bob chooses one of the pairs at random, and inspects the other 9 pairs to ensure that each pair indeed contains an identical number of marbles.

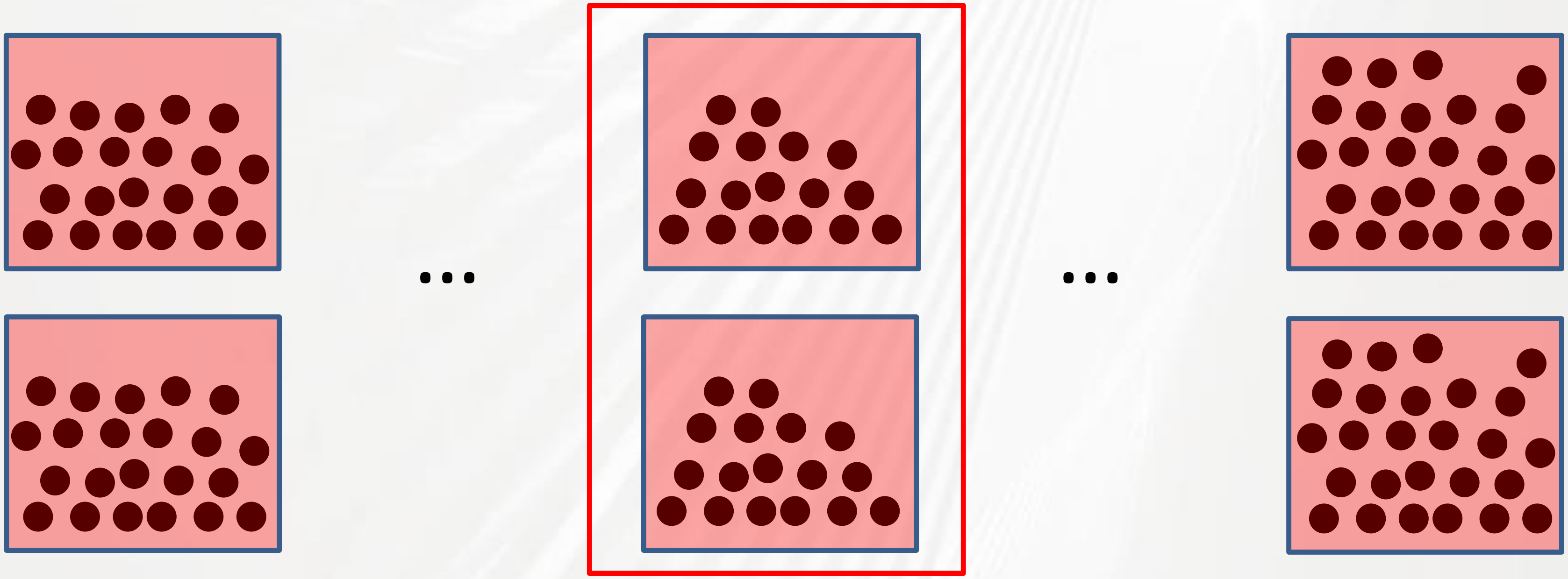
Zero-Knowledge Protocols – Equality Proof Example



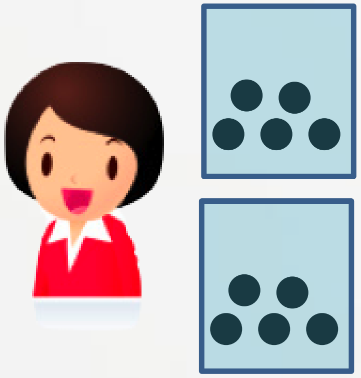
Alice has two cups each containing $x \in [0, n)$ marbles.

She wants to prove to Bob that both contain the same number without revealing x.

Alice prepares 10 pairs of buckets, both buckets in the i^{th} pair containing a random number $R_i \in [0, N)$ of marbles.



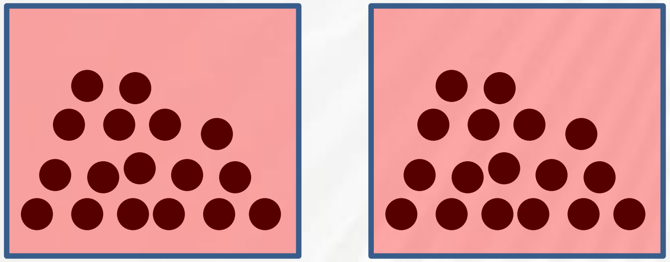
Zero-Knowledge Protocols – Example



Alice has two cups each containing $x \in [0, n)$ marbles.

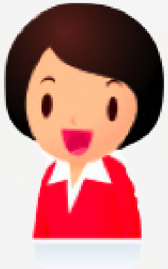
She wants to prove to Bob that both contain the same number without revealing x.

Alice pours the marbles from the first cup to the first bucket, and from the second cup to the second bucket.



Both contain $R_i \in_r [0, N)$ marbles

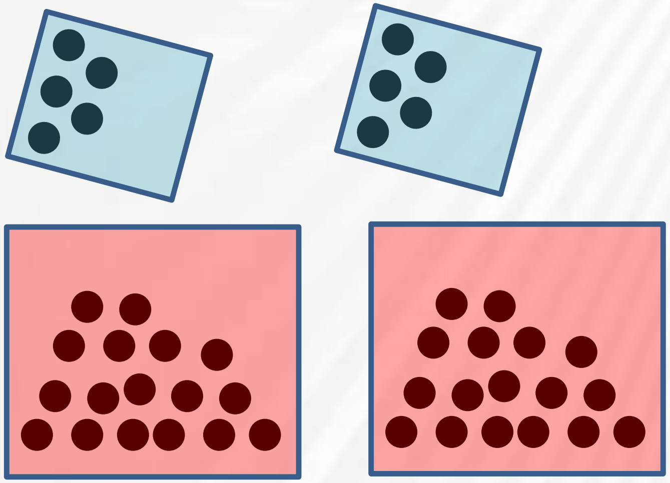
Zero-Knowledge Protocols – Equality Proof Example



Alice has two cups each containing $x \in [0, n)$ marbles.

She wants to prove to Bob that both contain the same number without revealing x.

Alice pours the marbles from the first cup to the first bucket, and from the second cup to the second bucket.



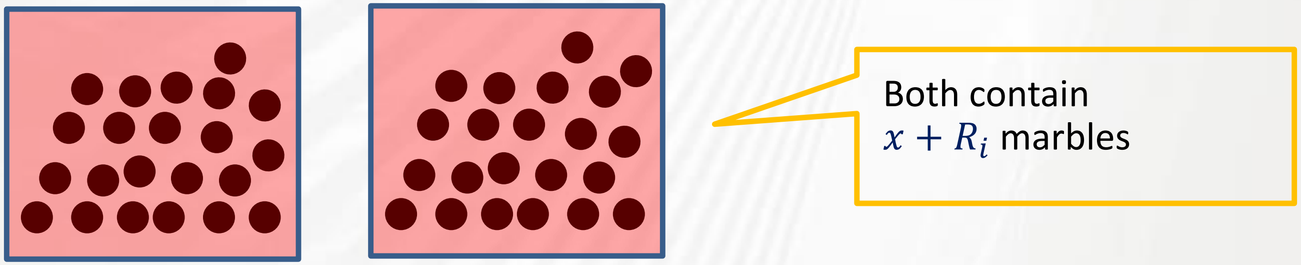
Both contain $R_i \in_r [0, N)$ marbles

Zero-Knowledge Protocols – Equality Proof Example

Alice has two cups each containing $x \in [0, n)$ marbles.

She wants to prove to Bob that both contain the same number without revealing x.

Alice pours the marbles from the first cup to the first bucket, and from the second cup to the second bucket.



Both contain $x + R_i$ marbles

Bob accepts the proof if both buckets contain the same number of marbles.

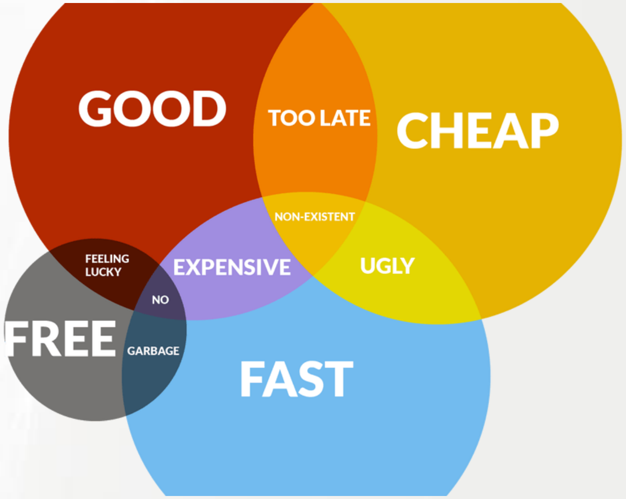
Soundness: If the cups contain a different number of marbles, Bob rejects with prob ≥ 0.9

Zero Knowledge: The number $x + R_i$ Bob sees is distributed n/N close to the uniform distribution on $(0, N]$. (Other 9 numbers are independent of X)

Credit to Boaz Barak ©

What is the success probability?

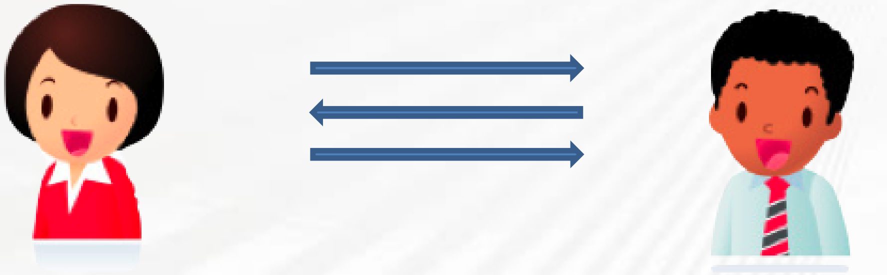
99,99..9 %



Properties of ZKP

□ Completeness:

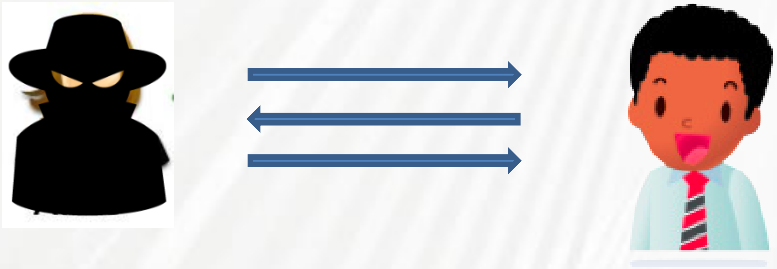
- if the statement is true, the honest verifier will be convinced of this fact by an honest prover.



→ accept

□ Soundness:

- if the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.

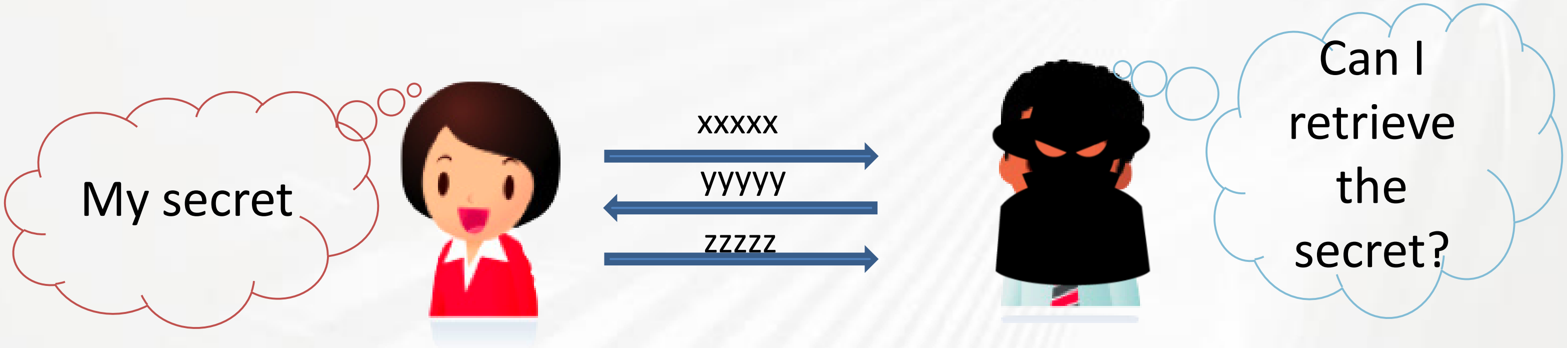


→ false

Properties of ZKP

❑ **Zero-knowledge**:

- if the statement is true, no verifier learns anything other than the fact that the statement is true.



My secret

xxxxx

yyyyy

zzzzz

Can I retrieve the secret?

Formalized by showing that every verifier has some *simulator* that, given only the statement to be proved (and no access to the prover), can produce a transcript that "looks like" an interaction between the honest prover and the verifier in question.

No secret

simulator

xxxxx

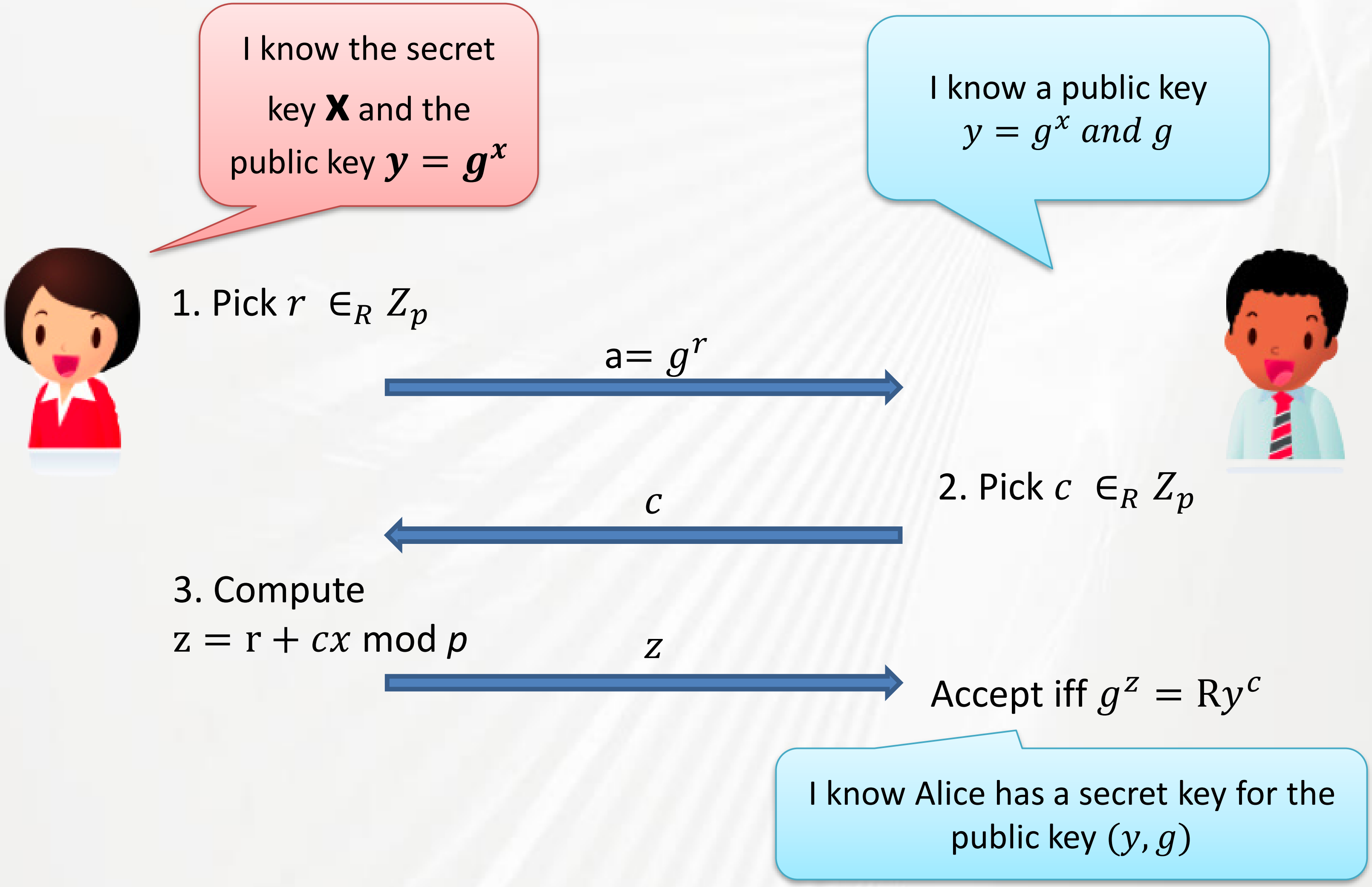
yyyyy

zzzzz

Zero-Knowledge Proof Schemes

Classical Schnorr Proofs	• C P Schnorr [1989] Efficient identification and signatures for smart cards, Crypto '89
zk-SNARKS	• E Ben-Sasson, A Chiesa, E Tromer, M Virza [2014] Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture. USENIX'14
zk-STARKS	• E Ben-Sasson, I Bentov, Y Horesh, M Riabzev [2018] Scalable, transparent, and post-quantum secure computational integrity. e-print 2018/046
Bulletproofs	• B Bünz, J Bootle, D Boneh et al [2018] : Bulletproofs: Short Proofs for Confidential Transactions and More IEEE S&P'18.

Simple ZK proof - Schnorr’s Protocol



I know the secret key $\boldsymbol{x}$ and the public key $\boldsymbol{y = g^x}$

I know a public key $y = g^x\ and\ g$

1. Pick $r\ \in_R Z_p$

a= g^r

2. Pick $c\ \in_R Z_p$

c

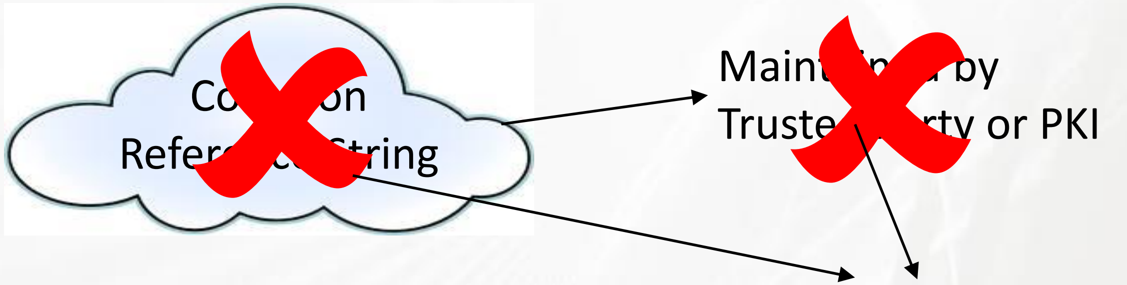
3. Compute $\mathrm{z} = \mathrm{r} + cx \bmod p$

z

Accept iff $g^z = \mathrm{R}y^c$

I know Alice has a secret key for the public key (y, g)

Variant: Non-Interactive ZK (NIZK)

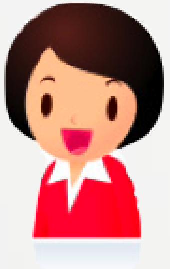


~~Common Reference String~~

~~Maintained by Trusted Party or PKI~~

Using Blockchains

I know the secret key **X** and the public key $\boldsymbol{y = g^x}$



1. Pick $r \in_R Z_p$, compute $R = g^r$

2. Pick $c = Hash(R, y, g)$

3. Compute $\mathrm{z} = \mathrm{r} + cx$ mod p

I know a public key $y = g^x$ *and* g

c, z

Compute $\mathrm{R} = g^z y^{-c}$

Accept iff $c = Hash(R, y, g)$

I know Alice has a secret key for the public key (y, g)

Confidential transaction in Monero

- The **amount confidentiality** is provided by using **Pedersen commitment**
  - The correctness **(= balance**) of the input and output amount is guaranteed by the **additive homomorphic** property of using Pedersen commitment.
  - But we still need to ensure that for every transaction amount M:

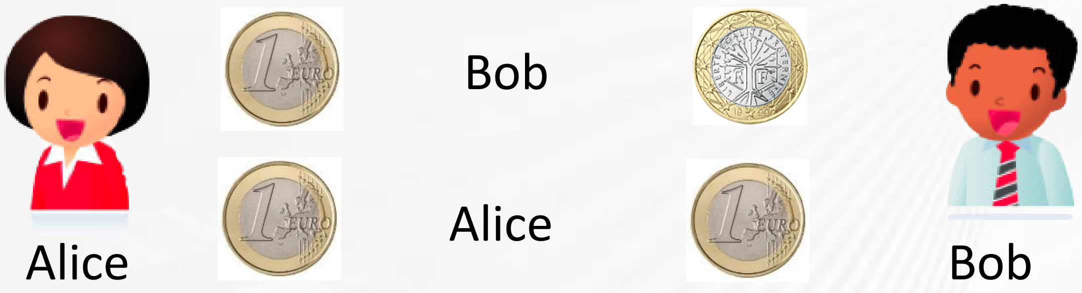
$$0 \le M < max$$

→ We need a (compact) **zero-knowledge range proof** for all transaction amount M!

⇩

- They use inner product argument (Bulletproof)
  - Represent each amount M as a binary vector $(a_1, a_2, \ldots, a_n)$
  - showed in ZK that M = $(a_1, a_2, \ldots, a_n)$ • $(1, 2, 4, 8, \ldots, 2^{k-1})$
  - → $0 \le M < 2^k$

Commitment Schemes



Alice

Bob

Alice

Bob

- Example:
  - Alice and Bob must agree who will clean tonight
  - They are at their offices. Each tosses a coin & they call:
    - If tosses are the same, then Alice cleans
    - If tosses are different, then Bob cleans
  - Who talks first?

Commitment Schemes

Alice

Alice

Bob

Bob

- ➢ Alice and Bob toss
  - Alice talks first

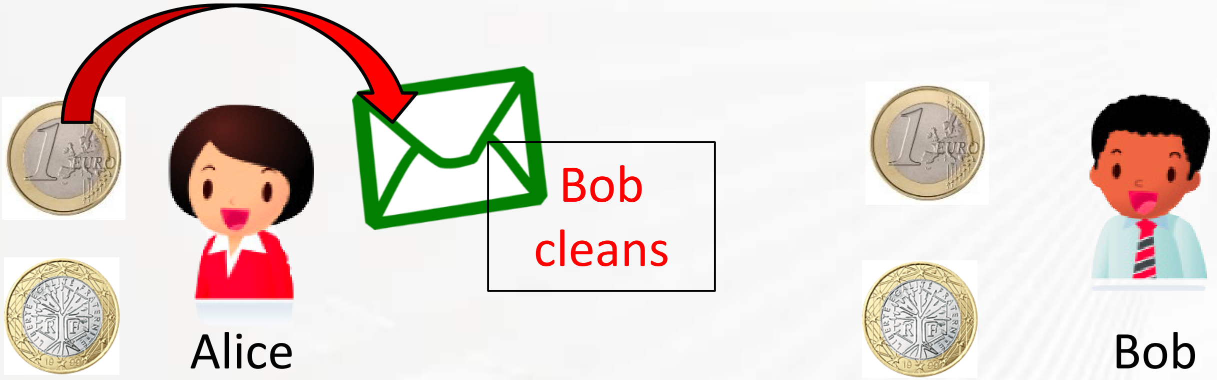
    Bob says he tossed the same value

  - Bob talks first

    Alice says she tossed the opposite value

- ➢ How can we avoid this?

Commitment Schemes



- Commitment: an envelope with a strange seal
  - Alice talks first
  - Commit phase: she hides toss in envelope, gives it to Bob
  - Bob reveals toss
  - Reveal phase: Alice tells Bob how to unseal envelope

Commitment Schemes

➢ Properties:

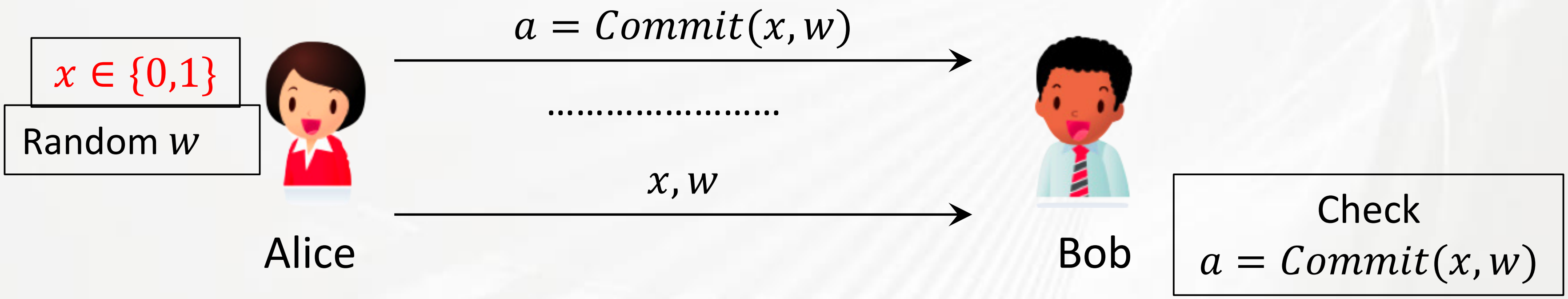
- Hiding: The content of the envelope is not visible

  Bob doesn't know anything about Alice's toss

- Binding: Alice can't change the content in the envelope

  Alice can't cheat after getting Bob's toss

Pedersen Commitments



$x \in \{0,1\}$

Random w

Alice

$a = Commit(x,w)$

……………………

x, w

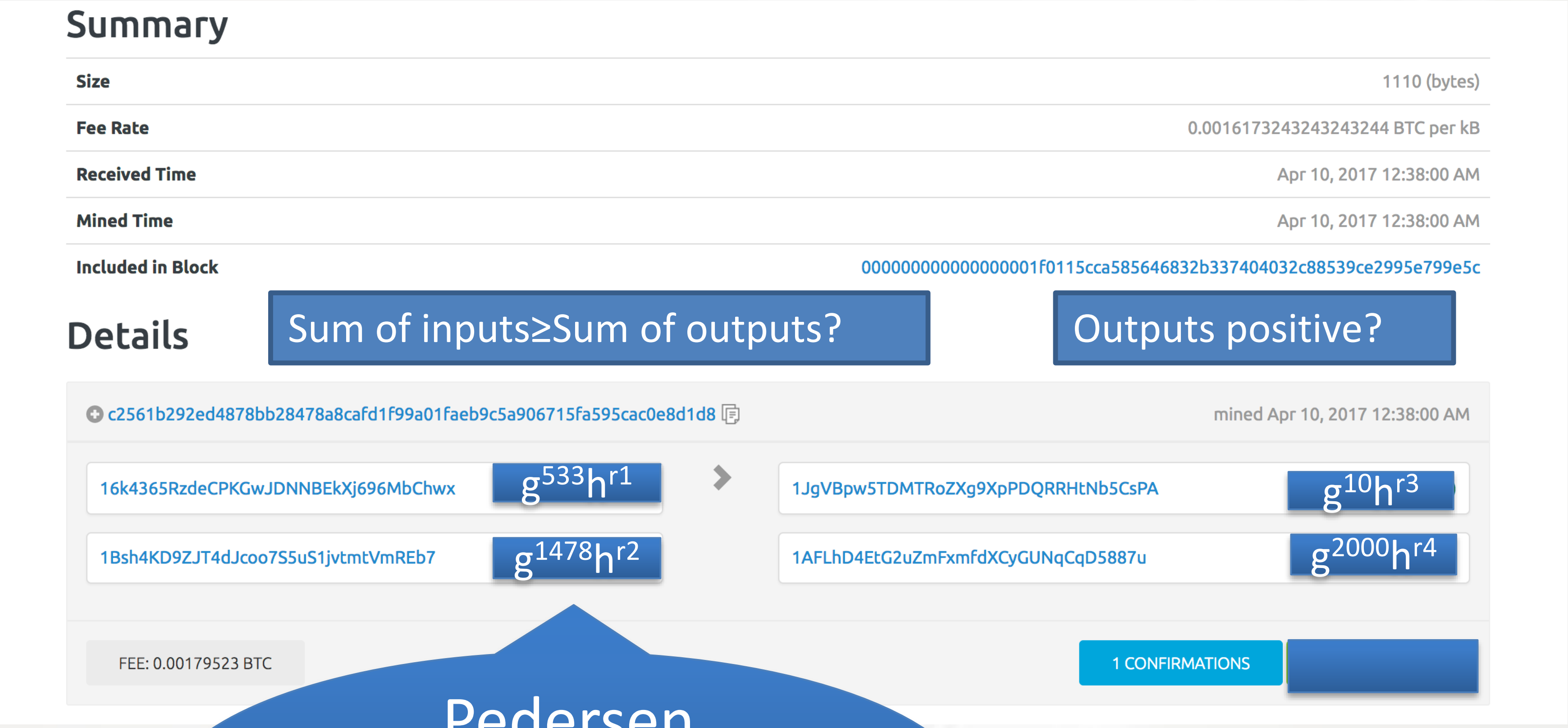
Bob

Check

$a = Commit(x,w)$

- Setup: $G_p^* = < g >$, prime field, $h = g^s \in G_p^* \setminus \{1\}$, s unknown
- Commitment of input value $x \in \{0,1\}$:
  - Choose random witness $w \leftarrow_R \{1, \ldots, p-1\}$
  - Compute $Commit(x,w) = g^w h^x = g^w g^{xs} = g^{w+xs}$
- Binding: Alice can’t change the content in the envelope? **Computational**
- Hiding: The content of the envelope is not visible ? **Info. Theoretical**

Confidential Transactions



Summary

Size	1110 (bytes)
Fee Rate	0.0016173243243243244 BTC per kB
Received Time	Apr 10, 2017 12:38:00 AM
Mined Time	Apr 10, 2017 12:38:00 AM
Included in Block	0000000000000000001f0115cca585646832b337404032c88539ce2995e799e5c

Details

Sum of inputs≥Sum of outputs?

Outputs positive?

c2561b292ed4878bb28478a8cafd1f99a01faeb9c5a906715fa595cac0e8d1d8

mined Apr 10, 2017 12:38:00 AM

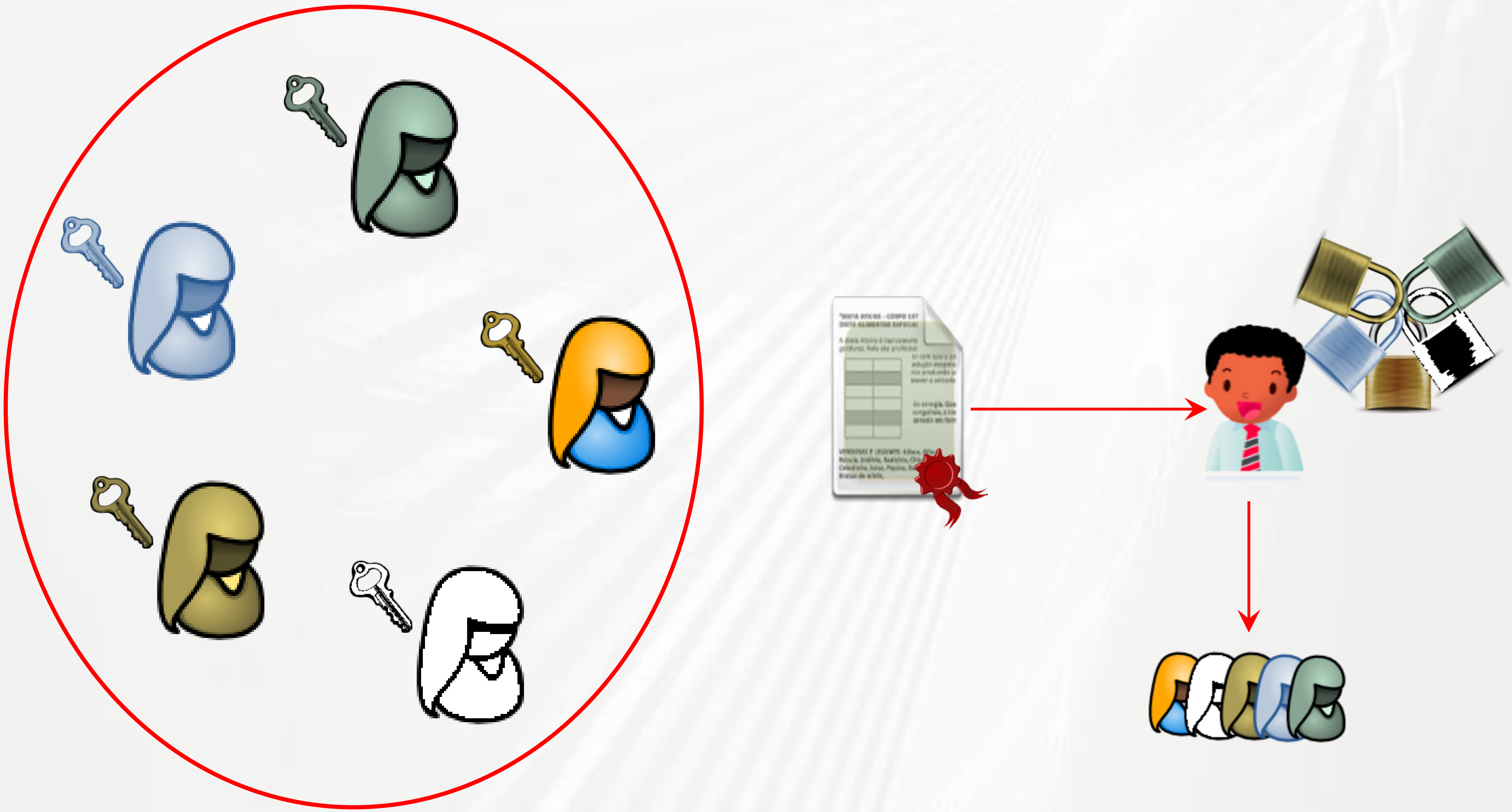
16k4365RzdeCPKGwJDNNBEkXj696MbChwx	$g^{533}h^{r1}$	1JgVBpw5TDMTRoZXg9XpPDQRRHtNb5CsPA	$g^{10}h^{r3}$
1Bsh4KD9ZJT4dJcoo7S5uS1jvtmtVmREb7	$g^{1478}h^{r2}$	1AFLhD4EtG2uZmFxmfdXCyGUNqCqD5887u	$g^{2000}h^{r4}$

FEE: 0.00179523 BTC

1 CONFIRMATIONS

Pedersen commitment: Commit(x;r)=g^xh^r

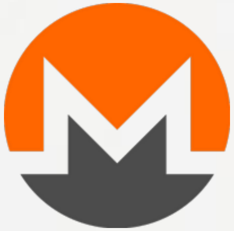
Ring Signatures



Credit to Cristina Onete ©

Projects using ZKPs

Bulletproofs



Use Bulletproofs for more efficient **range proofs only** and **not for privacy directly**

Bulletproofs (Range Proofs)

Proving that a number is within a range

$$\mathbf{v} \in [0,2^n)$$

Zero Knowledge about the Inner Product of Two Vectors

Any number can be represented as inner product of two vectors.

$$\mathbf{5 = <[1, 0, 1] , [2^2, 2^1, 2^0]>}$$

5 equals inner product of 2 vectors [1, 0, 1] and $[2^2, 2^1, 2^0]$

Bulletproofs (Range Proofs)

This is also how binary works

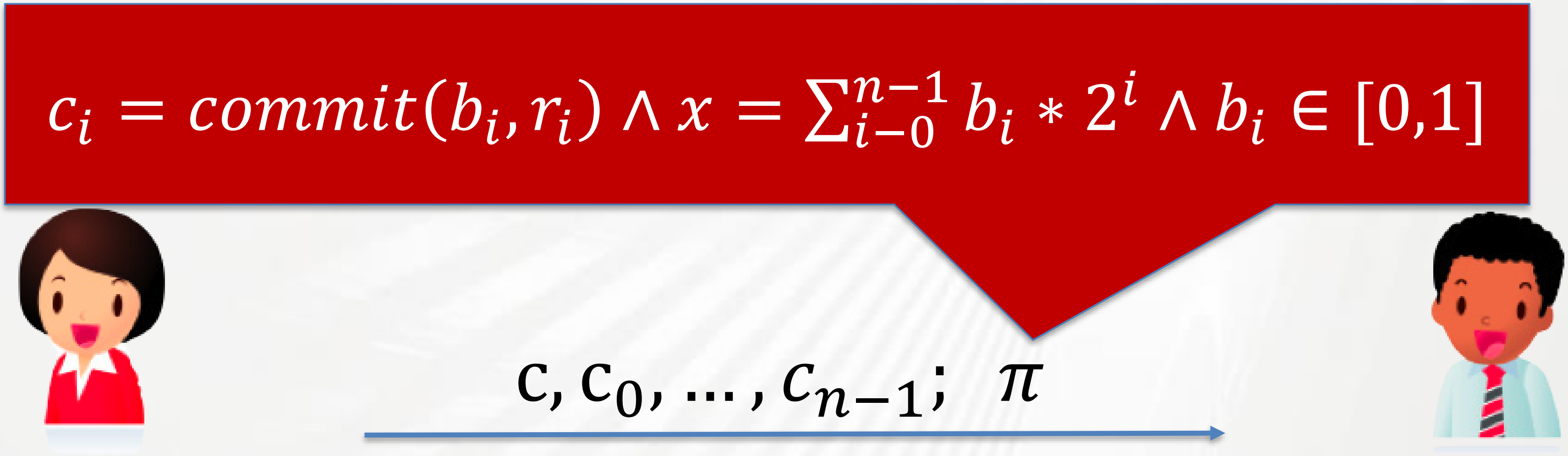
$101_{binary} = 5_{decimal}$ since $1(2^2) + 0(2^1) + 1(2^0)$

$v = \langle a, 2^n \rangle$

Example:
v = 5 and we wanted to prove that 5 is in range of 0 to 2^n **without showing 5**

$v \in [0,2^n)$

Concrete Range Proof using bit commitments



$$c_i = commit(b_i, r_i) \wedge x = \sum_{i-0}^{n-1} b_i * 2^i \wedge b_i \in [0,1]$$

$$\mathrm{c}, \mathrm{c}_0, \dots, c_{n-1}; \quad \pi$$

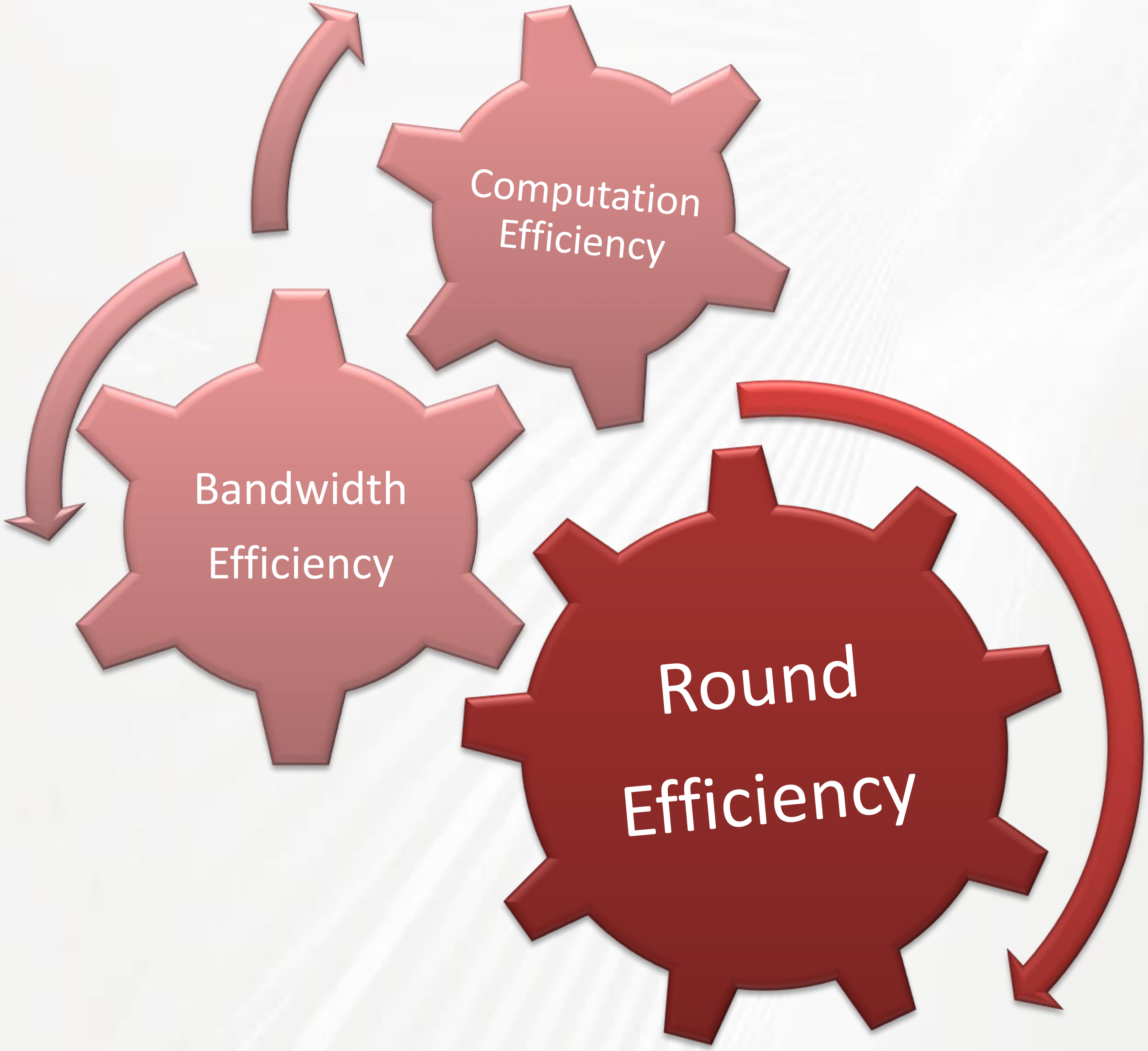
$$x = (b_0, \dots, b_{n-1}), b_i \in [0,1]$$

$$r_i \leftarrow \mathbb{Z}_q \forall\, i \in [0, n-1]$$

$$c_i = commit(b_i; r_i) \forall\, i \in [0, n-1]$$

Credit to Benedikt Bünz ©

Research Directions

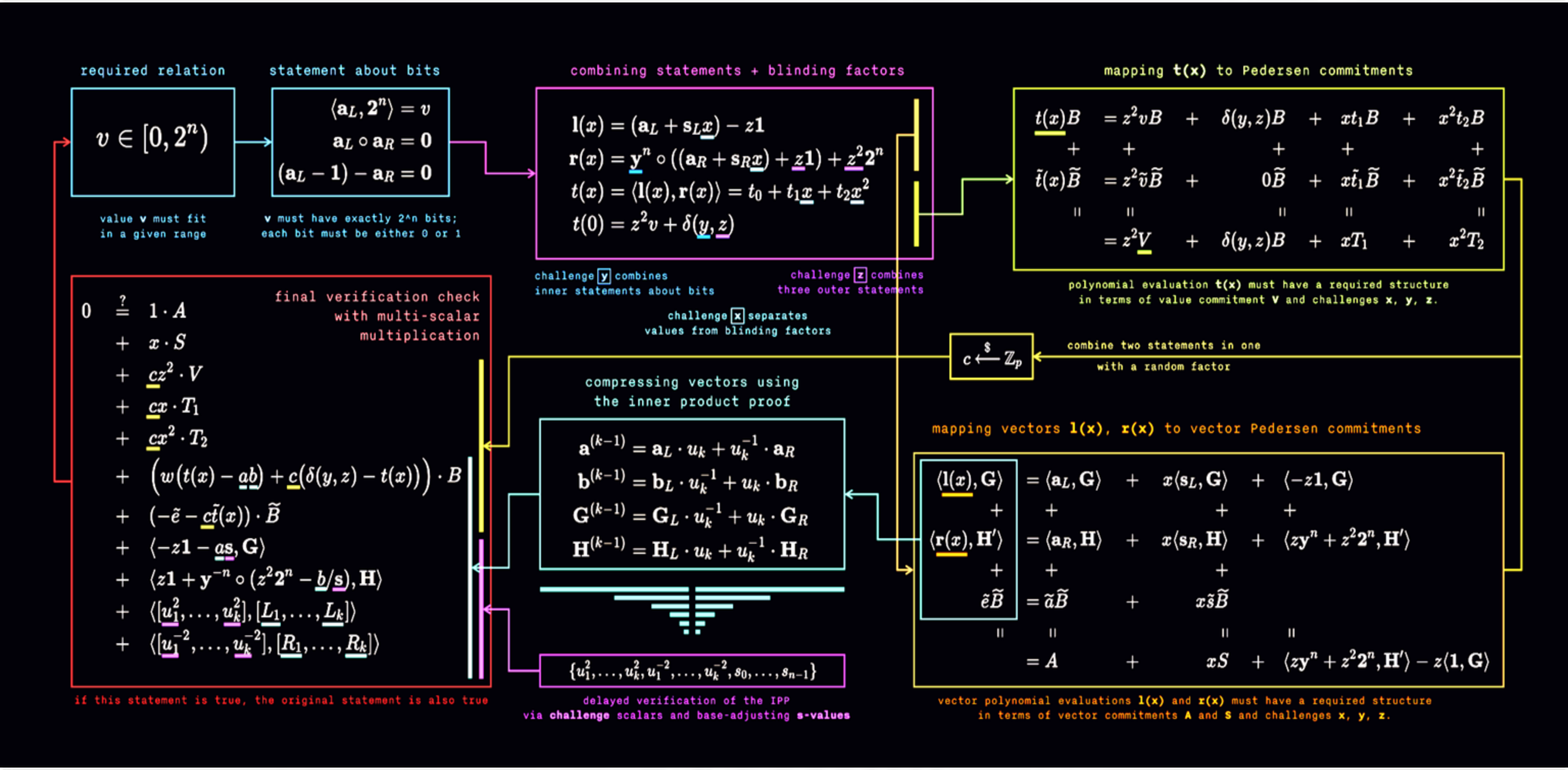


Computation Efficiency

Bandwidth Efficiency

Round Efficiency

Full Scheme of the Bullutproofs



required relation

$$v \in [0, 2^n)$$

value v must fit in a given range

statement about bits

$$\langle \mathbf{a}_L, \mathbf{2}^n \rangle = v$$
$$\mathbf{a}_L \circ \mathbf{a}_R = \mathbf{0}$$
$$(\mathbf{a}_L - \mathbf{1}) - \mathbf{a}_R = \mathbf{0}$$

v must have exactly 2^n bits; each bit must be either 0 or 1

combining statements + blinding factors

$$\mathbf{l}(x) = (\mathbf{a}_L + \mathbf{s}_L x) - z\mathbf{1}$$
$$\mathbf{r}(x) = \mathbf{y}^n \circ ((\mathbf{a}_R + \mathbf{s}_R x) + z\mathbf{1}) + z^2\mathbf{2}^n$$
$$t(x) = \langle \mathbf{l}(x), \mathbf{r}(x) \rangle = t_0 + t_1 x + t_2 x^2$$
$$t(0) = z^2 v + \delta(y, z)$$

challenge $\boxed{\mathbf{y}}$ combines inner statements about bits

challenge $\boxed{\mathbf{z}}$ combines three outer statements

challenge $\boxed{\mathbf{x}}$ separates values from blinding factors

mapping **t(x)** to Pedersen commitments

$$\begin{array}{cccccccccc}
t(x)B & = z^2 vB & + & \delta(y,z)B & + & xt_1B & + & x^2t_2B \\
+ & + & & + & & + & & + \\
\tilde{t}(x)\widetilde{B} & = z^2\tilde{v}\widetilde{B} & + & 0\widetilde{B} & + & x\tilde{t}_1\widetilde{B} & + & x^2\tilde{t}_2\widetilde{B} \\
\shortparallel & \shortparallel & & \shortparallel & & \shortparallel & & \shortparallel \\
 & = z^2 V & + & \delta(y,z)B & + & xT_1 & + & x^2T_2
\end{array}$$

polynomial evaluation **t(x)** must have a required structure in terms of value commitment **V** and challenges **x**, **y**, **z**.

$$c \xleftarrow{\$} \mathbb{Z}_p$$

combine two statements in one with a random factor

compressing vectors using the inner product proof

$$\mathbf{a}^{(k-1)} = \mathbf{a}_L \cdot u_k + u_k^{-1} \cdot \mathbf{a}_R$$
$$\mathbf{b}^{(k-1)} = \mathbf{b}_L \cdot u_k^{-1} + u_k \cdot \mathbf{b}_R$$
$$\mathbf{G}^{(k-1)} = \mathbf{G}_L \cdot u_k^{-1} + u_k \cdot \mathbf{G}_R$$
$$\mathbf{H}^{(k-1)} = \mathbf{H}_L \cdot u_k + u_k^{-1} \cdot \mathbf{H}_R$$

$$\{u_1^2, \dots, u_k^2, u_1^{-2}, \dots, u_k^{-2}, s_0, \dots, s_{n-1}\}$$

delayed verification of the IPP via **challenge** scalars and base-adjusting **s-values**

mapping vectors **l(x)**, **r(x)** to vector Pedersen commitments

$$\begin{array}{cccccccc}
\langle \mathbf{l}(x), \mathbf{G} \rangle & = \langle \mathbf{a}_L, \mathbf{G} \rangle & + & x\langle \mathbf{s}_L, \mathbf{G} \rangle & + & \langle -z\mathbf{1}, \mathbf{G} \rangle \\
+ & + & & + & & + \\
\langle \mathbf{r}(x), \mathbf{H}' \rangle & = \langle \mathbf{a}_R, \mathbf{H} \rangle & + & x\langle \mathbf{s}_R, \mathbf{H} \rangle & + & \langle z\mathbf{y}^n + z^2\mathbf{2}^n, \mathbf{H}' \rangle \\
+ & + & & + & & \\
\tilde{e}\widetilde{B} & = \tilde{a}\widetilde{B} & + & x\tilde{s}\widetilde{B} & & \\
\shortparallel & \shortparallel & & \shortparallel & & \shortparallel \\
 & = A & + & xS & + & \langle z\mathbf{y}^n + z^2\mathbf{2}^n, \mathbf{H}' \rangle - z\langle \mathbf{1}, \mathbf{G} \rangle
\end{array}$$

vector polynomial evaluations **l(x)** and **r(x)** must have a required structure in terms of vector commitments **A** and **S** and challenges **x**, **y**, **z**.

final verification check with multi-scalar multiplication

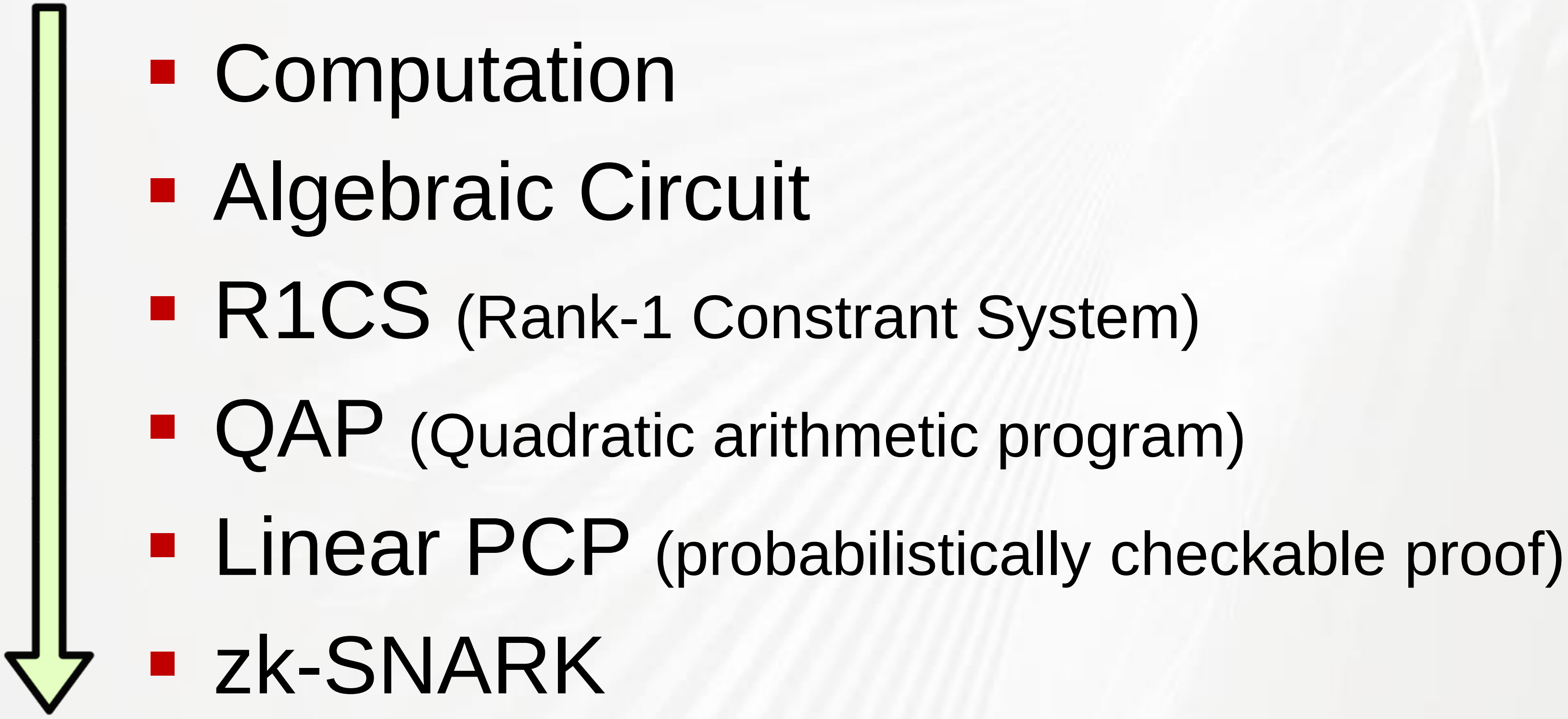
$$\begin{aligned}
0 \overset{?}{=}\ & 1 \cdot A \\
+\ & x \cdot S \\
+\ & cz^2 \cdot V \\
+\ & cx \cdot T_1 \\
+\ & cx^2 \cdot T_2 \\
+\ & \Big(w\big(t(x) - ab\big) + c\big(\delta(y,z) - t(x)\big)\Big) \cdot B \\
+\ & (-\tilde{e} - c\tilde{t}(x)) \cdot \widetilde{B} \\
+\ & \langle -z\mathbf{1} - a\mathbf{s}, \mathbf{G} \rangle \\
+\ & \langle z\mathbf{1} + \mathbf{y}^{-n} \circ (z^2\mathbf{2}^n - b/\mathbf{s}), \mathbf{H} \rangle \\
+\ & \langle [u_1^2, \dots, u_k^2], [L_1, \dots, L_k] \rangle \\
+\ & \langle [u_1^{-2}, \dots, u_k^{-2}], [R_1, \dots, R_k] \rangle
\end{aligned}$$

if this statement is true, the original statement is also true

https://github.com/dalek-cryptography/bulletproofs

B Bünz, J Bootle, D Boneh et al [2018] : Bulletproofs: Short Proofs for Confidential Transactions and More IEEE S&P’18.

zkSNARK construction via QAP and Linear PCPs



- Computation
- Algebraic Circuit
- R1CS (Rank-1 Constrant System)
- QAP (Quadratic arithmetic program)
- Linear PCP (probabilistically checkable proof)
- zk-SNARK

Credit to Eran Tromer ©

Properties of Zk-Snakrs

- Efficiency:
  - 288 byte **proof per transactions** (128-bit security)
  - <6 ms to **verify a proof**
  - <1 min to create

    for 2^{64} coins; asymptotically: log(#coins)
  - 896MB "system parameters"
    (fixed throughout system lifetime).

- **Trust in initial generation of system parameters (once).**

- Crypto assumptions:
  - Pairing-based elliptic-curve crypto
  - Less common: Knowledge of Exponent

    [Boneh Boyen 04] [Gennaro 04] [Groth 10] …
  - Properties of SHA256, encryption and signature schemes

Comparing Proof Systems (Oversimplified)

Proof System	Schnorr Σ-Protocol	Zk-SNARKs	STARKs	Bulletproofs
Proof Size	Long ✗	Very Short	Shortish	Shortish ✗
Prover	Linear	FFTs (memory req.)	FFT (Big memory req.) ✗	Multiexp.
Verifier	Linear ✗	Efficient	Efficient	Linear ✗
Trusted Setup	No	Required ✗	No	No
Practical	Yes	Yes	Not Quite ✗	Yes
Assumptions	Dlog + RO	Pairing +KoE	RO	Dlog + RO
Quantum Resistancy	No	No	Yes ✓	No

TÜBİTAK
BİLGEM
UEKAE

BLOKZİNCİR
Araştırma Laboratuvarı

TEŞEKKÜRLER

Dr. Muhammed Ali BİNGÖL

TÜBİTAK BİLGEM
National Research
Institute of Electronics and Cryptology
Blockchain Reseach Lab.

muhammedali.bingol@tubitak.gov.tr
T: +90 262 648 1702