Privacy, Anonymity, and Digital Currencies

Ian Miers
What do we mean by Anonymity?

**Anonymous**

/əˈnɒnɪməs/  •

*adjective*

- (of a person) not identified by name; of unknown name.
- *the anonymous author of Beowulf*
- *synonyms: unnamed, of unknown name, nameless, incognito, unidentified, unknown, unsourced, secret*
- *antonyms: known, identified, signed*
- *having no outstanding, individual, or unusual features; unremarkable or impersonal. “the anonymous black car waiting to take him to the airport”*
- *synonyms: characterless, nondescript, impersonal, faceless “an anonymous housing development”*
- *used in names of support groups for addicts of a substance or behavior to indicate the confidentiality maintained among members of the group. “Alcoholics Anonymous”*

**Origin**

Greek: an- *without* → Greek: ἄνων (anōn) *nameless* → Late Latin: _anonymus_ (from _an_- + _onom_)*name* → English: anonymous (late 16th century) via Late Latin from Greek _anōnemos_ ‘nameless’ (from _an_-‘without’ + _onoma_‘name’).
pseudonym
/'soʊdənɪm/

noun
noun: pseudonym; plural noun: pseudonyms

a fictitious name, especially one used by an author.
synonyms: pen name, nom de plume, assumed name, false name, alias, professional name, sobriquet, stage name, nom de guerre
"Geisel was best known by the pseudonym 'Dr. Seuss'

Origin
GREEK
pseudēs false

GREEK pseudōnymos

FRENCH pseudonyme pseudonym

early 19th century: from French pseudonyme, from Greek pseudōnymos, from pseudēs 'false' + onuma 'name.'
Pseudonymity vs anonymity

• People have pseudonyms on social media.
• We don’t know who they belong to
• Why do they make “new” throw away accounts to post more private things?
The ideal currency: Decentralized
Privacy for payments: by example

• Four scenarios for buying something online in terms of who learns what:
  1. No one learns anything (aka “anonymous”)
  2. Merchant learns who you are, but no one else does
     • Shipping packages to your house with cash paid on delivery.
  3. Only the merchant and the payment processor learn you bought something
     • Credit cards
     • Checks
  4. Everyone sees every transaction you make
     • ???
Bitcoin: Decentralized

Secure

Private?
Bitcoin privacy limitations? An example: buying two ebooks
Bitcoin privacy limitations?

Address

$3.1415

$2.718

$3.1415

$2.718
Address$_1$ $\rightarrow$ $\text{Price}_1$ $= \$3.1415$

Address$_1$ $\rightarrow$ $\text{Price}_2$ $= \$2.718$
Employees can learn salaries

Address_0

Address_a

Address_b

Address_c
Pseudonyms can be linked together

An Analysis of Anonymity in the Bitcoin System
F. Reid and M. Harrigan
PASSAT 2013

A Fistful of Bitcoins: Characterizing Payments Among Men with No Names
S. Meiklejohn et al.
IMC 2013
Network-layer de-anonymization

“The first node to inform you of a transaction is probably the source of it”

Dan Kaminsky
Black Hat 2011 talk
When the cookie meets the blockchain: Privacy risks of web payments via cryptocurrencies

Steven Goldfeder*, Harry Kalodner*, Dillon Reisman†, Arvind Narayanan*
Princeton University
*{stevenag, kalodner, arvindn}@cs.princeton.edu
†dillon@lonlon.io

Abstract—We show how third-party web trackers can deanonymize users of cryptocurrencies. We present two distinct but complementary attacks. On most shopping websites, third party trackers receive information about user purchases for purposes of advertising and analytics. We show that, if the user pays using a cryptocurrency, trackers typically possess enough information about the purchase to uniquely identify the transaction on the blockchain, link it to the user’s cookie, and further to the user’s real identity. Our second attack shows that if the tracker is able to link two purchases of the same user to the blockchain in this manner, it can identify the user’s entire cluster of addresses and transactions on the blockchain, even if the user employs blockchain anonymity techniques such as CoinJoin. The attacks are passive and hence can be retroactively applied to past purchases. We discuss several mitigations, but none are perfect.

I. INTRODUCTION

Eight years after Bitcoin’s introduction, the ability to pay online using cryptocurrencies is common: prominent merchants such as Microsoft, Newegg, and Overstock support it. Cryptocurrency users tend to value financial privacy, and it is a major reason for choosing to pay with Bitcoin [1]. Yet, websites including shopping sites are known to be rife with advertising and analytics. We show that trackers are able to combine user cookie information with blockchain transaction data to deanonymize users, even if they use a privacy-preserving technique such as CoinJoin. We further show that this deanonymization can be retroactively applied to past purchases.

Of course, Bitcoin does not guarantee unlinkability of transactions. But while linking of a user’s Bitcoin addresses with each other is well known [3]–[6], our attack shows how to link addresses to external information, including identity.

The main defense against linkage attacks is mixing [7], [8]. The best known mixing technique is CoinJoin, in which users send coins to each other in a way that hides the link between their old and new coins. Our second main contribution is showing the effectiveness of the cluster intersection attack, a previously known attack against mixing. Specifically, we show that a small amount of additional information, namely that two (or more) transactions were made by the same entity, is sufficient to undo the effect of mixing (see Figure 1). While such auxiliary information is available to many potential entities — merchants, other counterparties such as websites that accept donations, intermediaries such as payment processors, and potentially network eavesdroppers — web trackers are in the ideal position to carry out this attack.
Commercial companies are doing analysis.
Bitcoin:
Decentralized

Secure
Private
Twitter for your bank account

All your spending made public to:

• Your creepy ex
• Your business competitors
• EVERYBODY
Should we have anonymous money

- Money laundering
- Extortion
- Rhino poaching (according to 2 op-eds)

Last week, an op-ed that I wrote for The Baltimore Sun prompted a lot of very strong reactions, both positive and negative. I argued that efforts to make Bitcoins functionally anonymous are very dangerous, because money laundering is inherently very dangerous.

To summarize my argument: transnational crime is a global business valued in the hundreds of billions of dollars, and criminals need a way to easily launder, move, and invest that money to...
Criminals already have anonymity

- Hard part of money laundering isn’t anonymizing your money, it’s making it have legitimate sources on the other side
- You can hide what you are doing with cash.
• Just because the government has a right to know what you are doing doesn’t mean everyone should

• Companies need confidentiality to conduct business

• Fungibility is a necessary property of money
E-cash

Requires:
1. No link between withdraw and spend
2. No double spending
Techniques for anonymity

• Centralized Mixing
• Decentralized mixing
• Non-interactive mixing
• Fully anonymous payments
Bitcoin Mix/laundry

The simplest way to a little anonymity
Bitcoin Laundries

Decentralized

Secure

Private
Trying to break the link
Mix
Can get coins back to yourself
Mix
Requires Trust

Mix can steal funds

Mix can log transactions
Caution: Mixing services may themselves be operating with anonymity. As such, if the mixing output fails to be delivered or access to funds is denied there is no recourse. Use at your own discretion.

— Bitcoin Wiki
Bitcoin Laundries

Decentralized

Secure

Private?
Decentralized Mixing

Removing the trusted party
Why decentralized mixing?

- No bootstrapping problem
- Theft impossible
- Possibly better anonymity
- More philosophically aligned with Bitcoin
Coinjoin

Each signature is entirely separate

This is 1 mixing round

Mixing principles from before apply on top of basic protocol

Proposed by Greg Maxwell, Bitcoin core developer

August 2013
Coinjoin algorithm

1. Find peers who want to mix
2. Exchange input/output addresses
3. Construct transaction
4. Send it around, collect signatures (Before signing, each peer checks if her output is present)
5. Broadcast the transaction
Coinjoin: remaining problems

- How to find peers
- Peers know your input-output mapping (This is a worse problem than for centralized mixes)
- Denial of service
Finding peers

Use an untrusted server
Peer anonymity

Strawman solution:
1. exchange inputs
2. disconnect and reconnect over Tor
3. exchange outputs

Better solution:
special-purpose anonymous routing mechanism
Denial of service

Proposed solutions:

• Proof of work
• Proof of burn
• Server kicks out malicious participant
• Cryptographic “blame” protocol

(CoinShuffle: Practical Decentralized Coin Mixing for Bitcoin
T. Ruffing et al., PETS 2014)
Coinjoin

Generate a single transaction requiring All parties to sign

Single transaction mixes all parties to gether

Due to Bitcoin core developer Greg Maxwell
Coinjoin (continued)

1. Find peers to mix with
2. Anonymously exchange input and output addresses
3. Construct tx
4. Each user signs TX after insuring it pays them
5. Broadcast transaction
Advantages: trustless

• No theft risk: Each party only signs the transaction and puts their funds in if the transaction also pays them
• No tracking: if peers anonymously exchange addresses, no one knows where the coins go.
Challenges

• Easy to disrupt since every party must behave correctly
• Peers must be online to mix
Challenges

• Still need enough peers online to mix with
• Anonymity is limited to who is online
• Dos and reliability issues are still a concern even with mitigations
• Payments aren’t instant: must wait for mixing
Decentralized Mixing

Decentralized

Secure

Private?
Down the cryptographic Rabbit Hole

Improving cryptocurrencies through cryptography
Commitments

• Cryptographically opaque envelope
• Content cannot be read by anyone else
• Cannot be changed by anyone

\[ \text{Comm}(x; r) = g^x h^r \]
Pedersen Commitments

\[ Comm(x; r) = g^x h^r \]

- Information theoretically blinding
- hiding under discrete log assumption
- Homomorphic:

\[
Comm(x; r) \cdot (Comm(y; r')) = g^x h^r \cdot g^y h^{r'} \\
= g^{x+y} h^{r+r'} \\
= Comm(x + y; r + r')
\]

\[
Comm(x; r)/(Comm(y; r')) = g^x h^r / g^y h^{r'} \\
= g^{x-y} h^{r-r'} \\
= Comm(x - y; r - r')
\]

May also see it as \( xG + rH \)
Zero-knowledge proofs

- Zero-knowledge [Goldwasser, Micali, & Rackof 1985]
- Lets you make statements about the content of commitments
- Sound: cannot be forged
- Zero knowledge: can keep secrets
So what?

What do we do with all of this crypto
Aren't you a little short for a stormtrooper?
Confidential Transactions (Maxwell 2015)

- Instead of storing value in the clear in each transaction, commit to it.
- Prove that the sum of the value in the output commitments is less than the sum of the input commitments.

```
<table>
<thead>
<tr>
<th>To:</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>7</td>
</tr>
<tr>
<td>Alice</td>
<td>5</td>
</tr>
<tr>
<td>Alice</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Inputs</td>
<td>outputs</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Values:
- Bob: 7
- Alice: 5 + 5 = 10
- Alice: 3

Input commitments: 5 + 5 = 10
Output commitments: 7 + 3 = 10

TX: Value (total): 10
Confidential Transactions (Maxwell 2015)

- Instead of storing value in the clear in each transaction, commit to it
- Prove that the sum of the value in the output commitments is less than the sum of the input commitments.

Inputs

To: Alice
Value: Comm(5)

To: Bob
Value: Comm(7)

TX

Outputs

π

To: Alice
Value: Comm(5)

To: Alice
Value: Comm(7)
Confidential Transactions (Maxwell 2015)

• Instead of storing value in the clear in each transaction, commit to it.
• Prove that the sum of the value in the output commitments is less than the sum of the input commitments.
Non-interactive mixing

Mix with people without them knowing it
Cryptonote/Monero

- Original idea by Van Saberhagen Oct 2013***
- Improved on by Monero with RingCT
- Deployed in Monero (April 2014)
- Effectively a non-interactive mix + confidential transactions
Interactive and non interactive mixes

Decentralized

Secure

Private?
(a) Bitcoin

(b) Cryptonote
Chain mixes to increase privacy: theory
Chain mixes to increase privacy: theory
Chain mixes to increase privacy: theory
Chain mixes to increase privacy:
Chain mixes to increase privacy: theory
Chain mixes to increase privacy: theory
When the cookie meets the blockchain: Privacy risks of web payments via cryptocurrencies

Steven Goldfeder*, Harry Kalodner*, Dillon Reisman†, Arvind Narayanan*
Princeton University
*{stevenag, kalodner, arvindn}@cs.princeton.edu
†dillon@lonlon.io

Abstract—We show how third-party web trackers can deanonymize users of cryptocurrencies. We present two distinct but complementary attacks. On most shopping websites, third party trackers receive information about user purchases for purposes of advertising and analytics. We show that, if the user pays using a cryptocurrency, trackers typically possess enough information about the purchase to uniquely identify the transaction on the blockchain, link it to the user’s cookie, and further to the user’s real identity. Our second attack shows that if the tracker is able to link two purchases of the same user to the blockchain in this manner, it can identify the user’s entire cluster of addresses and transactions on the blockchain, even if the user employs blockchain anonymity techniques such as CoinJoin. The attacks are passive and hence can be retroactively applied to past purchases. We discuss several mitigations, but none are perfect.

I. INTRODUCTION

Eight years after Bitcoin’s introduction, the ability to pay online using cryptocurrencies is common: prominent merchants such as Microsoft, Newegg, and Overstock support it. Cryptocurrency users tend to value financial privacy, and it is a major reason for choosing to pay with Bitcoin [1]. Yet, websites including shopping sites are known to be rife with tracking mechanisms. Web browsers send cookies from web pages even if it is not leaked to them by default. We show that this is possible on the vast majority of merchant sites.

Of course, Bitcoin does not guarantee unlinkability of transactions. But while linking of a user’s Bitcoin addresses with each other is well known [3]–[6], our attack shows how to link addresses to external information, including identity.

The main defense against linkage attacks is mixing [7], [8]. The best known mixing technique is CoinJoin, in which users send coins to each other in a way that hides the link between their old and new coins. Our second main contribution is showing the effectiveness of the cluster intersection attack, a previously known attack against mixing. Specifically, we show that a small amount of additional information, namely that two (or more) transactions were made by the same entity, is sufficient to undo the effect of mixing (see Figure 1). While such auxiliary information is available to many potential entities — merchants, other counterparties such as websites that accept donations, intermediaries such as payment processors, and potentially network eavesdroppers — web trackers are in the ideal position to carry out this attack.
Anonymous (e)cash

Strong privacy
Digitizing money

- Two ways to do it:
  - Create digital cash
  - Create digital checks
Digital Checks
Digital cash

- Sign with private key to prevent counterfeiting
- Problem: double spend attack
- Can make single use currency
  - Get a unique serial number when you withdraw money
  - Spend it by showing an unused serial number
- Can be made anonymous using Blind Signatures
Zero-knowledge-Succinct Non-Interactive Arguments of Knowledge

- Constant size + fast to verify
- Proof statements is boolean or arithmetic circuit
- Need new crypto to work efficiently in circuits
  - hash based commitments
    - \text{comm}=\text{SHA256}(\text{value} \| \text{randomness})
  - Merkle trees for set membership
Centralized e-cash via signatures
Centralized Anonymous e-cash via blind signatures
E-cash+Bitcoin:
Decentralized

Secure
Private
E-cash + Bitcoin
E-cash+Bitcoin:

Decentralized

Secure

Private
Zero coin

May 2013

Ian Miers, Christina Garman, Matthew Green, Aviel Rubin
Zerocoin

\[ \pi \in \{ \text{42} \} \]
Naïve zero knowledge proof

• Given a list of coins $C_1, C_2, \ldots, C_n$ proof
  • I know the commitment $C$ to a serial number $S$
  • The commitment is either $C_1$ OR $C_2$ OR $\ldots$

• Proof hides which coin is being spent because it is zero-knowledge.

• Proof prevents coin forgery because it is sound.
Problem with OR Proof

• Size of the proof: O(n)
• Computational work : O(n)
• You need to know all of the coins every made to make it

• This is what’s used by some post Zerocoin currencies: e.g. Cryptonote/Monero.
  • Anonymity set per tx is about 5 coins per tx
Zerocoin: basic approach

• Use a compact set membership proof to show that $c \in \{c_1, c_2, ..., c_n\}$

• Trick is finding a set membership proof that is compact and works with efficient zero-knowledge proofs.

• Use an RSA accumulator.
• Strong RSA accumulator originally due to Benaloh and de Mare
• Efficient proof for accumulation of primes proposed by Camenisch and Lysyanskaya ’01
• Factorization of $N$ must be secret

Define set as: $$A = u^{c_1 \cdot c_2 \cdot c_n} \mod N$$

Witness to membership: $$w_i = u^{c_1 \cdot c_2 \cdot c_{i-1} \cdot c_{i+1} \cdot \ldots \cdot c_n} \mod N$$

Verify witness as: $$w_i^{c_i} \equiv A \mod N$$
Limitations of Zerocoin

• Proof is rather large: 25kb.
• Fixed value payments.
• Only allows you to anonymize your own coins
Zero cash

January/May 2014

Eli Ben-Sasson, Alessandro Chiesa,
Christina Garman, Matthew Green, Eran Tromer,
Ian Miers, and Madars Virza
Working with zkSNARKs

- Hash functions work better with circuit representations than 1024+ bit groups.
- Use hash based commitments \( \text{comm} = \text{SHA256}(\text{value} || |r\text{andomness}) \)
- Use Merkle tree’s for membership proof instead of accumulator
From Zerocoin to Zerocash

• With more power, what could we do?
• Commitments are general purpose.
  • Store coin value.
  • Store owner
Zerocash

• Each coin is a commitment to
  • The public key of the owner
  • The value of the coin
  • Some randomness
Zerocash Challenges

• Still need the serial number for double spending
• Now serial number can be generated adversarially
  • Can deanonymize users
  • Can potentially steal funds
  • Can DOS spends by colliding serial numbers
Solution:

- Create an address with a public key and private key
- Commitment includes public key
- Must prove knowledge of private key to spend funds
- Serial number is a function of sender selected randomness AND receiver private key
Zerocash

\[ \text{Merkle Tree Root} = H(\text{child}_{\text{left}} || \text{child}_{\text{right}}) \]

... up to \(2^{64}\) coin commitments

internal structure of a coin commitment

coin serial number
Zerocash: Alice Pays Bob $3

<table>
<thead>
<tr>
<th>Serial #’s</th>
<th>31</th>
<th>29</th>
<th>34</th>
<th>11</th>
<th>7</th>
<th>8</th>
<th>6</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coins</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
<td>👉</td>
</tr>
</tbody>
</table>

blockchain
# Zerocash: Alice Pays Bob $3

<table>
<thead>
<tr>
<th>Serial #’s</th>
<th>31</th>
<th>29</th>
<th>34</th>
<th>11</th>
<th>7</th>
<th>8</th>
<th>6</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coins</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Alice: $10

Cin₁

Bob: $3

Cout₁
Zerocash: Alice Pays Bob $3

Alice Pays Bob $3

<table>
<thead>
<tr>
<th>Serial #’s</th>
<th>31</th>
<th>29</th>
<th>34</th>
<th>11</th>
<th>7</th>
<th>8</th>
<th>6</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coins</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Zerocash: Alice Pays Bob $3

Serial #’s

| 31 | 29 | 34 | 11 | 7  | 8  | 6  | 17 |

Coins

| Alice $10 |
| Bob $3 |

$Cin_1$

$Cout_1$
Zerocash: Alice Pays Bob $3

blockchain

Serial #’s            | 31 | 29 | 34 | 11 | 7  | 8  | 6  | 17 |

Coins

Cin₁

Alice $10

Bob $3

Cout₁

Alice $7

Cout₂
Zerocash: Alice Pays Bob $3

Proof:
- $\text{Cin}_1$ is:
  - In Merkel tree $R$
  - addressed to Alice
  - has Serial numbers 42
- $\text{Cin}_1 - \text{Cout}_1 - \text{Cout}_2 \geq 0$
- $\text{Cout}_1$ and $\text{Cout}_2$ are properly formed

<table>
<thead>
<tr>
<th>Serial #'s</th>
<th>(31)</th>
<th>(29)</th>
<th>(34)</th>
<th>(11)</th>
<th>(7)</th>
<th>(8)</th>
<th>(6)</th>
<th>(17)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coins</td>
<td>$\text{Alice} \quad $10\quad$</td>
<td>$\text{Bob} \quad $3\quad$</td>
<td>$\text{Alice} \quad $7\quad$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Zerocash: Alice Pays Bob $3

Proof:
- \( \text{Cin}_1 \) is:
  - In Merkel tree \( R \)
  - Addressed to Alice
  - Has Serial numbers 42

- \( \text{Cin}_1 - \text{Cout}_1 - \text{Cout}_2 \geq 0 \)
- \( \text{Cout}_1, \text{Cout}_2 \) are properly formed

<table>
<thead>
<tr>
<th>Serial #’s</th>
<th>Coins</th>
</tr>
</thead>
<tbody>
<tr>
<td>31 29 34 11 7 8 6 17</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
</tbody>
</table>

Serial #'s

Coins

blockchain

42

R

\( \pi \)

\( \text{Cout}_1 \)

\( \text{Cout}_2 \)
Zerocash: Alice Pays Bob $3

Proof:
- Cin₁ is:
  - In Merkel tree R
  - Addressed to Alice
  - Has serial numbers 42
- Cin₁ - Cout₁ - Cout₂ ≥ 0
- Cout₁, Cout₂ are properly formed

Serial #’s

| 31 | 29 | 34 | 11 | 7  | 8  | 6  | 17 |

Coins

| 💌 | 💌 | 💌 | 💌 | 💌 | 💌 | 💌 | 💌 |

blockchain

42 64  R

Zerocash:
Alice Pays Bob $3

π

Cout₁

Cout₂
Zcash limitations

- zkSNARKs requires trusted setup
- Compromised trusted set up can lead to counterfeit money
- CANNOT compromise privacy
- Currently privacy preserving transactions in zcash are expensive (take 40s and 3GB of work) and therefor optional
- Recent work has gotten that down to 7 seconds and 40mb.s
Zerocash:
Decentralized
Secure
Private
(a) Bitcoin

(b) Cryptonote

(c) Zcash