MixCoin: Anonymity for BitCoin with Accountable Mixes

Bonneau, Narayanan, Miller, Clark, Kroll, and Felten

Presented by A. Wood

The Graduate Center, CUNY

May 27, 2016

BitCoin

- Decentralized digital currency

BitCoin

- Decentralized digital currency
- Worth over $6 billion

BitCoin

- Decentralized digital currency
- Worth over $6 billion
- Uses a public, distributed ledger to log transactions

BitCoin

- Decentralized digital currency
- Worth over $6 billion
- Uses a public, distributed ledger to log transactions
- Pseudonymous

Psuedononymity in BitCoin

- Does not provide true anonymity

Psuedononymity in BitCoin

- Does not provide true anonymity
- Users have pseudononymous addresses

Psuedononymity in BitCoin

- Does not provide true anonymity
- Users have pseudononymous addresses
- Transactions can often be easily linked

Psuedononymity in BitCoin

- Does not provide true anonymity
- Users have pseudononymous addresses
- Transactions can often be easily linked
- If one transaction is linked to user, then all of their addresses may be exposed

Background: BitCoin

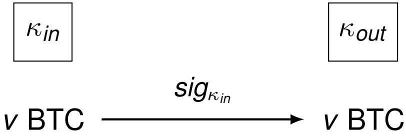
- An address κ is a public key

Background: BitCoin

- An address κ is a public key
- Addresses are psuedonanymous

Background: BitCoin

- An address κ is a public key
- Addresses are psuedonanymous
- BitCoin transaction:



Background: BitCoin

- Transactions are recorded on the blockchain

Background: BitCoin

- Transactions are recorded on the blockchain
- Blockchain is a decentralized, publicly verifiable ledger

Background: BitCoin

- Transactions are recorded on the blockchain
- Blockchain is a decentralized, publicly verifiable ledger
- Records all past messages exchanged between users on that Blockchain

Background: BitCoin

- Transactions are recorded on the blockchain
- Blockchain is a decentralized, publicly verifiable ledger
- Records all past messages exchanged between users on that Blockchain
- No transactions are truly anonymous, because they are always publicly visible on the blockchain

Background: BitCoin Mixes

Used to preserve privacy for some BitCoin users. Multiple clients send coins to a mixing address, which forwards them randomly to a fresh address for each client.

Mixing Services

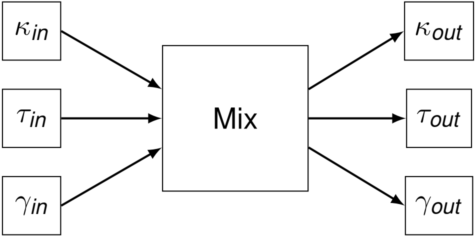
Mixing Services

- Take a user's coins and randomly exchange them for other user's coins

Mixing Services

- Take a user's coins and randomly exchange them for other user's coins
- Obfuscates ownership

Alice owns N bitcoins at address κ_{in}, which is linkable to her real-world identity. She wishes to tranfer her funds to address κ_{out} in a way which is difficult to link to κ_{in} for a fee.
She sends her funds to a mix M which holds them for an agreed time period before sending an equal value to κ_{out}.



The Downside of Mixes

- Slow mixtime

The Downside of Mixes

- Slow mixtime
- Low transaction volume

The Downside of Mixes

- Slow mixtime
- Low transaction volume
- Open question of linking attacks between inputs and outputs

The Downside of Mixes

No protection from theft!

The Downside of Mixes

- A malicious mix could send Alice's money to its own address instead of Alice's

The Downside of Mixes

- A malicious mix could send Alice's money to its own address instead of Alice's
- Alice could falsely accuse the mix of theft to undermine its reputation

The Downside of Mixes

- A malicious mix could send Alice's money to its own address instead of Alice's
- Alice could falsely accuse the mix of theft to undermine its reputation
- Accusations of theft cannot be proven, hence it is difficult to determine which mixes are honest

The Downside of Mixes

- A malicious mix could send Alice's money to its own address instead of Alice's
- Alice could falsely accuse the mix of theft to undermine its reputation
- Accusations of theft cannot be proven, hence it is difficult to determine which mixes are honest
- A malicious mix is able to link the in and out addresses, potentially undermining Alice's anonymity

Various Solutions

- ZeroCoin

Various Solutions

- ZeroCoin
  - Provides strong anonymity
  - Requires advanced cryptography
  - Substantial modifications to BitCoin

Various Solutions

- ZeroCoin
    - Provides strong anonymity
    - Requires advanced cryptography
    - Substantial modifications to BitCoin
- Zerocash

Various Solutions

- ZeroCoin
  - Provides strong anonymity
  - Requires advanced cryptography
  - Substantial modifications to BitCoin
- Zerocash
  - Entirely New Currency

Various Solutions

- ZeroCoin
  - Provides strong anonymity
  - Requires advanced cryptography
  - Substantial modifications to BitCoin
- Zerocash
  - Entirely New Currency
- CoinJoin, CoinSwap

Various Solutions

- ZeroCoin
  - Provides strong anonymity
  - Requires advanced cryptography
  - Substantial modifications to BitCoin
- Zerocash
  - Entirely New Currency
- CoinJoin, CoinSwap
  - Backwards-compatible with BitCoin
  - Practical complications, smaller anonymity sets

Solution: MixCoin!

- Build on the framework of mixes

Solution: MixCoin!

- Build on the framework of mixes
- Add cryptographic accountability layer

MixCoin: Accountability

- MixCoin mixes issues signed warranties

MixCoin: Accountability

- MixCoin mixes issues signed warranties
- If Alice sends v coins by time t_1, then the mix sends v coins back to her by time t_2

MixCoin: Accountability

- MixCoin mixes issues signed warranties
- If Alice sends v coins by time t_1, then the mix sends v coins back to her by time t_2
- Alice can publish this warranty if the mix fails to deliver her coin

MixCoin: Randomized mixing fees

- Paying for mixing services incentives honest behavior

MixCoin: Randomized mixing fees

- Paying for mixing services incentives honest behavior
- A fixed fee can undermine anonymity in multiple mixing

MixCoin: Randomized mixing fees

- Paying for mixing services incentives honest behavior
- A fixed fee can undermine anonymity in multiple mixing
- MixCoin uses randomized, all-or-nothing fees

MixCoin: Mix indistinguishability

- Single-use mix addresses

MixCoin: Mix indistinguishability

- Single-use mix addresses
- Passive adversaries can't determine which mix a user is interacting with

MixCoin: Mix indistinguishability

- Single-use mix addresses
- Passive adversaries can't determine which mix a user is interacting with
- Anonymity set: All users who are interacting with any mix at the same time

MixCoin: Mix networks for BitCoin

- Chains multiple mixes together

MixCoin: Mix networks for BitCoin

- Chains multiple mixes together
- Provides strong anonymity against an active attacker who can break mix indistinguishability

MixCoin

- Mixing completes in a few hours

MixCoin

- Mixing completes in a few hours
- Mixing fees less than 1%

MixCoin

- Mixing completes in a few hours
- Mixing fees less than 1%
- Can be deployed immediately on top of BitCoin

MixCoin: The Idea

- MixCoin is a protocol for mixing with accountability

MixCoin: The Idea

- MixCoin is a protocol for mixing with accountability
- The mix gives Alice a signed warranty which she can use to unabiguously prove that the mix has misbehaved

MixCoin: The Idea

- MixCoin is a protocol for mixing with accountability
- The mix gives Alice a signed warranty which she can use to unabiguously prove that the mix has misbehaved
- There is no way to prove that a mix is not storing records which could deanonymize its clients

MixCoin: The Idea

- MixCoin is a protocol for mixing with accountability
- The mix gives Alice a signed warranty which she can use to unabiguously prove that the mix has misbehaved

- There is no way to prove that a mix is not storing records which could deanonymize its clients
- Alice can send her coins through a series of mixes which all must collude to deanonymize her final address

Assumptions

- Availability of multiple mixes M_i

Assumptions

- Availability of multiple mixes M_i
- Mix M_i represented by warranty-signing key K_{M_i}

Assumptions

- Availability of multiple mixes M_i
- Mix M_i represented by warranty-signing key K_{M_i}
- Each mix's warranty-signing key is used consistently

Assumptions

- Availability of multiple mixes M_i
- Mix M_i represented by warranty-signing key K_{M_i}
- Each mix's warranty-signing key is used consistently
- Alice able to negotiate with mix over an anonymous, confidential channel (Tor hidden service)

Mixing Parameters

- v, the value to be mixed

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix
- t_2, the deadline by which the mix must return funds to Alice

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix
- t_2, the deadline by which the mix must return funds to Alice
- κ_{out}, the address where Alice is transferring her funds

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix
- t_2, the deadline by which the mix must return funds to Alice
- κ_{out}, the address where Alice is transferring her funds
- ρ, the mixing fee

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix
- t_2, the deadline by which the mix must return funds to Alice
- κ_{out}, the address where Alice is transferring her funds
- ρ, the mixing fee
- n, a nonce used to determine payment of randomized mixing fees

Mixing Parameters

- v, the value to be mixed
- t_1, the deadline by which Alice must send funds to the mix
- t_2, the deadline by which the mix must return funds to Alice
- κ_{out}, the address where Alice is transferring her funds
- ρ, the mixing fee
- n, a nonce used to determine payment of randomized mixing fees
- w, the number of blocks the Mix requires to confirm Alice's payment

Mixing Parameters

- Note: the value v is a standardized “chunk" size which the mix accepts.

Mixing Parameters

- Note: the value v is a standardized “chunk" size which the mix accepts.
- Deadlines are specified as block numbers in the BitCoin block chain.

Mixing Parameters

- Note: the value v is a standardized “chunk" size which the mix accepts.
- Deadlines are specified as block numbers in the BitCoin block chain.
- $w = 6$ is a common standard

Step 1

Alice contacts Mix over a secure channel and proposes the mixing parameters

Step 2

There are two cases:

Step 2

There are two cases:

1. The mix accepts these terms, generates a fresh escrow address κ_{esc}, and sends back a warranty containing all of Alice's parameters plus κ_{esc}

Step 2

There are two cases:

1. The mix accepts these terms, generates a fresh escrow address κ_{esc}, and sends back a warranty containing all of Alice's parameters plus κ_{esc}
2. The mix rejects Alice's request

Note that κ_{out} and κ_{esc} should be fresh addresses created specifically for mixing

Step 3

Alice transfers the value v to κ_{esc} by time deadline t_1

Step 4

1. The mix transfers an equal value to κ_{out} by time t_2

Step 4

1. The mix transfers an equal value to κ_{out} by time t_2
2. The mix fails to transfer v to κ_{out} by time t_2

Step 5

1. If the protocol is successful, *A* and *M* destroy their records

Step 5

1. If the protocol is successful, A and M destroy their records
2. If Alice detects theft, she publicizes

$$\{v, t_1, t_2, w, \kappa_{esc}, \kappa_{out}, \rho, n\}_{K_M}$$

Mixing Fees

Fixed mixing fees undermine the goal of indistinguishable transfers and limit the anonymity set

Randomized Mixing Fees

- With probability ρ the mix retains the entire value v. With probability $1 - \rho$ the mix takes no fee at all.

Randomized Mixing Fees

- With probability ρ the mix retains the entire value v. With probability $1 - \rho$ the mix takes no fee at all.
- Expected mixing rate is ρ

Randomized Mixing Fee

- Must use a publicly verifiable mechanism to randomly choose which chunks to retain as mixing fees

Randomized Mixing Fee

- Must use a publicly verifiable mechanism to randomly choose which chunks to retain as mixing fees
- Call this a *beacon*

Randomized Mixing Fee

- Must use a publicly verifiable mechanism to randomly choose which chunks to retain as mixing fees
- Call this a *beacon*
- This computation can be performed by anybody if Alice's warranty is published (hence cheating is detectable!)

The Beacon

- May be external to Bitcoin

The Beacon

- May be external to Bitcoin
  - e.g., NIST's beacon, financial data

The Beacon

- May be external to Bitcoin
  - e.g., NIST's beacon, financial data
- Randomness may be extracted from future BitCoin blocks

The Beacon

- May be external to Bitcoin
  - e.g., NIST's beacon, financial data
- Randomness may be extracted from future BitCoin blocks
  - Assuming the exact set of future transactions is included in each block

The Beacon

- May be external to Bitcoin
    - e.g., NIST's beacon, financial data
- Randomness may be extracted from future BitCoin blocks
    - Assuming the exact set of future transactions is included in each block
    - Also utilizes the nonce *n* specified by Alice, used to solove the proof-of-work puzzle

Beacon from BitCoin blocks

The mix computes

$$X = \text{Beacon}(t_1, w, n) = \text{PRNG}(n \| B_{t_1+w}) \xleftarrow{R} (0, 1)$$

Beacon from BitCoin blocks

The mix computes

$$X = \text{Beacon}(t_1, w, n) = \text{PRNG}(n \| B_{t_1+w}) \xleftarrow{R} (0, 1)$$

- PRNG is a cryptographic pseudo random function which outputs a value uniformly drawn from the range $(0, 1)$

Beacon from BitCoin blocks

The mix computes

$$X = \text{Beacon}(t_1, w, n) = \text{PRNG}(n \| B_{t_1+w}) \xleftarrow{R} (0, 1)$$

- PRNG is a cryptographic pseudo random function which outputs a value uniformly drawn from the range $(0, 1)$
- B_i is the Merkle root of block i in the BlockChain

Merkle Trees

- double-SHA256 (SHA256 applied twice)

Merkle Trees

- double-SHA256 (SHA256 applied twice)

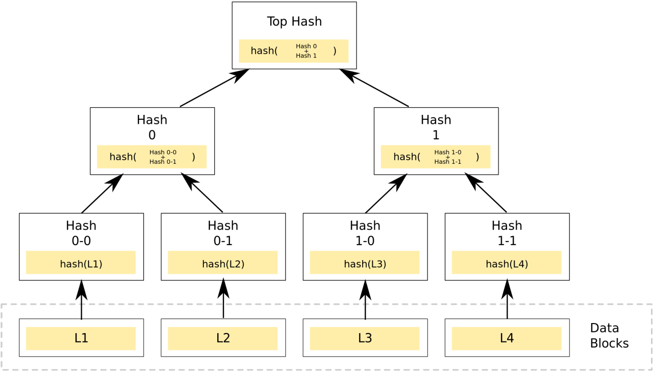


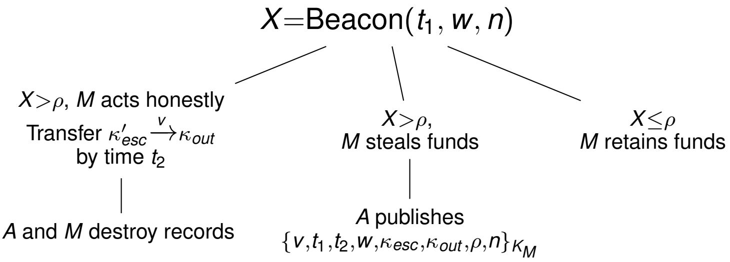
Figure: From Wikipedia

The MixCoin Protocol

$A \xrightarrow{anon} M$
$\langle v, t_1, t_2, w, \kappa_{out}, \rho, n \rangle$

- *M* accepts terms
  $A \xleftarrow{anon} M$
  $\{v, t_1, t_2, w, \kappa_{esc}, \kappa_{out}, \rho, n\}_{K_M}$
  - *A* by time t_1:
    Transfer $\kappa_{in} \xrightarrow{v} \kappa_{esc}$
    - $X = \text{Beacon}(t_1, w, n)$
  - *M* aborts protocol $\perp$
- *M* rejects terms
  $A \xleftarrow{anon} M$
  $\perp$
  - *A* destroys κ_{out}

The MixCoin Protocol



Mixing Fees to Miners

What if the miners would like a fee? Say they want to be paid τ BTC.

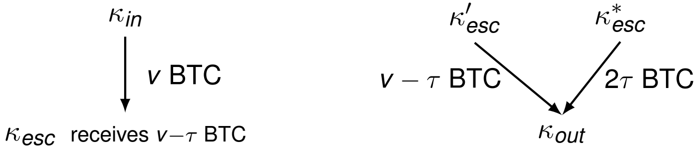
Mixing Fees to Miners

What if the miners would like a fee? Say they want to be paid τ BTC.

κ^*_{esc} is a third address from which the mix previously retained a mixing fee.

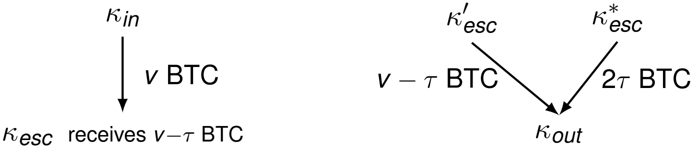
Mixing Fees to Miners

What if the miners would like a fee? Say they want to be paid τ BTC.



Mixing Fees to Miners

What if the miners would like a fee? Say they want to be paid τ BTC.



κ^*_{esc} is a third address from which the mix previously retained a mixing fee.

Sequential Mixing

What if Alice wants to send her funds through N independent mixes?

Sequential Mixing

What if Alice wants to send her funds through N independent mixes?

- Alices chooses a sequence of mixes $M_1, M_2, \ldots, M_N$

Sequential Mixing

What if Alice wants to send her funds through N independent mixes?

- Alices chooses a sequence of mixes $M_1, M_2, \ldots, M_N$
- Execute the MixCoin protocol through these mixes in reverse order

Sequential Mixing

What if Alice wants to send her funds through N independent mixes?

- Alices chooses a sequence of mixes $M_1, M_2, \ldots, M_N$
- Execute the MixCoin protocol through these mixes in reverse order
- Instruct M_i to forward her funds to escrow address $\kappa_{esc_{i+1}}$ which she previously received from M_{i+1}

Sequential Mixing

What if Alice wants to send her funds through N independent mixes?

- Alices chooses a sequence of mixes $M_1, M_2, \ldots, M_N$
- Execute the MixCoin protocol through these mixes in reverse order
- Instruct M_i to forward her funds to escrow address $\kappa_{esc_{i+1}}$ which she previously received from M_{i+1}
- Alice obtains N signed warranties, transfers funds to κ_{esc_1}

Sequential Mixing

- Alice most likely wants to transfer kv BTC.

Sequential Mixing

- Alice most likely wants to transfer *kv* BTC.
- She must negotiate a total of *kN* warranties with mixes.

Sequential Mixing

- Alice most likely wants to transfer kv BTC.
- She must negotiate a total of kN warranties with mixes.
- Each chunk should travel through an independently-chosen random mix of sequences.

Mix Incentives

- Mix fees incentivize honest behavior in mixes

Mix Incentives

- Mix fees incentivize honest behavior in mixes
- Higher fees more strongly incentivize honesty

Mix Incentives

- Mix fees incentivize honest behavior in mixes
- Higher fees more strongly incentivize honesty
- Users should avoid mixes charging less than some minimum value ρ

Mix Incentives

- Mix fees incentivize honest behavior in mixes
- Higher fees more strongly incentivize honesty
- Users should avoid mixes charging less than some minimum value ρ
- What is this ρ?

Mix Incentives

- Mix has two choices at any given block in time: continue or abscond

Mix Incentives

- Mix has two choices at any given block in time: continue or abscond
- Q is the average amount of money flowing into the mix during one block

Mix Incentives

- Mix has two choices at any given block in time: continue or abscond
- Q is the average amount of money flowing into the mix during one block
- $\bar{t}$ is the average time period (in blocks) that the mix holds funds during a mixing round

Mix Incentives

- Mix has two choices at any given block in time: continue or abscond
- Q is the average amount of money flowing into the mix during one block
- $\bar{t}$ is the average time period (in blocks) that the mix holds funds during a mixing round
- Expected value of absconding is $\mathbf{E}[\text{abscond}] = Q\bar{t}$

Mix Incentives

- Expected payoff from choosing to continue defined recursively

Mix Incentives

- Expected payoff from choosing to continue defined recursively
- Under steady state conditions, optimal decision the same in every round

Mix Incentives

- Expected payoff from choosing to continue defined recursively
- Under steady state conditions, optimal decision the same in every round
- If mix chooses to continue, it will do so indefinitely

Mix Incentives

- Expected payoff from choosing to continue defined recursively
- Under steady state conditions, optimal decision the same in every round
- If mix chooses to continue, it will do so indefinitely
- Mix is discounting future earnings at a rate of r per block

Mix Incentives

- Expected payoff from choosing to continue defined recursively
- Under steady state conditions, optimal decision the same in every round
- If mix chooses to continue, it will do so indefinitely
- Mix is discounting future earnings at a rate of r per block
- Net value of indefinite honest behavior:

$$\frac{\rho Q}{r}$$

Mix Incentives

- To incentivize honest behavior, we need:

$$\frac{\rho}{r} > \bar{t}$$

Mix Incentives

- To incentivize honest behavior, we need:

$$\frac{\rho}{r} > \bar{t}$$

- Let r be equivalent to the highest available risk-free rate of return available

Mix Incentives

- To incentivize honest behavior, we need:

$$\frac{\rho}{r} > \bar{t}$$

- Let r be equivalent to the highest available risk-free rate of return available
- Then, all this says is that the expected value of fees collected by a mix during the time it holds funds is greater than the amount those funds would yield during the same time period if invested

Mix Incentives

- Low mixing fees should incentivize honest behavior

Mix Incentives

- Low mixing fees should incentivize honest behavior
- If $r \approx 20\%$ available to mix, then a mix time $\bar{t} \approx 1$ hour yields lower bound $\rho_{min} \approx 2^{-15}$

Mix Incentives

- Low mixing fees should incentivize honest behavior
- If $r \approx 20\%$ available to mix, then a mix time $\bar{t} \approx 1$ hour yields lower bound $\rho_{min} \approx 2^{-15}$
- A chunk taking a path through 10 consecutive mixes leaves a fee rate of $\approx 2^{-12}$

Anonymity Properties: Passive Adversary

- Best case scenario: Passive Adversary

Anonymity Properties: Passive Adversary

- Best case scenario: Passive Adversary
- Say adversary can determine with high probability which Bitcoin transactions are mix traffic

Anonymity Properties: Passive Adversary

- Best case scenario: Passive Adversary
- Say adversary can determine with high probability which Bitcoin transactions are mix traffic
- Adversary still may not be able to link escrow addresses to specific mixes due to their one-time nature

Anonymity Properties: Passive Adversary

- Best case scenario: Passive Adversary
- Say adversary can determine with high probability which Bitcoin transactions are mix traffic
- Adversary still may not be able to link escrow addresses to specific mixes due to their one-time nature
- This property is called *mix indistinguishability*

Anonymity Properties: Active Adversary

First attack:

Anonymity Properties: Active Adversary

First attack:

- When Alice sends a chunk from κ_{in} to the mix via κ_{esc}, the client who ultimately receives this chunk will learn that κ_{in} interacted with M.

Anonymity Properties: Active Adversary

First attack:

- When Alice sends a chunk from κ_{in} to the mix via κ_{esc}, the client who ultimately receives this chunk will learn that κ_{in} interacted with *M*.
- The client which sends the chunk to κ'_{esc}, eventually sent to κ_{out}, learns that Alice interacted with *M*

Anonymity Properties: Active Adversary

First attack:

- When Alice sends a chunk from κ_{in} to the mix via κ_{esc}, the client who ultimately receives this chunk will learn that κ_{in} interacted with *M*.
- The client which sends the chunk to κ'_{esc}, eventually sent to κ_{out}, learns that Alice interacted with *M*
- Active adversary can exploit this in a *flooding attack*

Anonymity Properties: Active Adversary

Second attack:

Anonymity Properties: Active Adversary

Second attack:

- If mixes pay transaction fees, then *M* may use a fee retained from a user to pay a the transaction fees

Anonymity Properties: Active Adversary

Second attack:

- If mixes pay transaction fees, then *M* may use a fee retained from a user to pay a the transaction fees
- All of these transaction fees can then be linked to *M*

Mixing Multiple Chunks

- If Alice combines many mixed chunks to make a payment, her anonymity set will be reduced to the intersection of the anonymity sets of all chunks

Mixing Multiple Chunks

- If Alice combines many mixed chunks to make a payment, her anonymity set will be reduced to the intersection of the anonymity sets of all chunks
- If she mixed those chunks sufficiently, they will have the same anonymity set

Mixing Multiple Chunks

- If Alice combines many mixed chunks to make a payment, her anonymity set will be reduced to the intersection of the anonymity sets of all chunks
- If she mixed those chunks sufficiently, they will have the same anonymity set
- If even one chunk travels through a path consisting entirely of comprimised mixes, her entire payment loses anonymity

Mixing Multiple Chunks

- If Alice combines many mixed chunks to make a payment, her anonymity set will be reduced to the intersection of the anonymity sets of all chunks
- If she mixed those chunks sufficiently, they will have the same anonymity set
- If even one chunk travels through a path consisting entirely of comprimised mixes, her entire payment loses anonymity
- If 25% of mixes are comprimised, there is a 2^{-20} chance of routing a chunk through a chain of ten comprimised mixes

Acknowledgements

I would like to thank Dr. Rosario Gennaro for teaching this course.

Bibliography

Mixcoin: Anonymity for BitCoin with Accountable Mixes, Joseph Bonneau, Arvind Narayanan, Andrew Miller, Jeremy Clark, Joshua A. Kroll, Edward W. Felten. International Financial Cryptography Association 2014, LCNS 8437, pp.486-504.