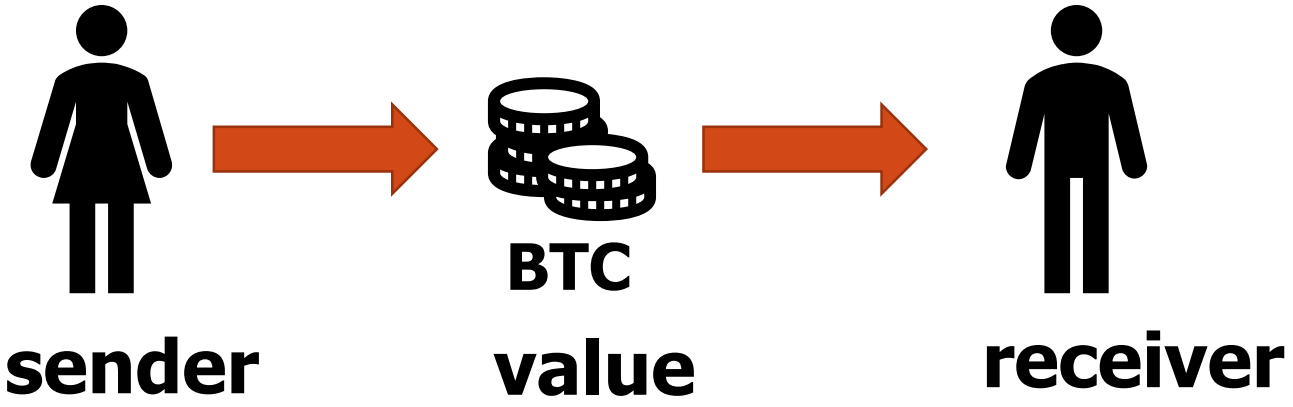
Improving the Mixin Sampling Algorithm for Better Untraceability in Monero Blockchain

Yujie Pan
2018.6.1

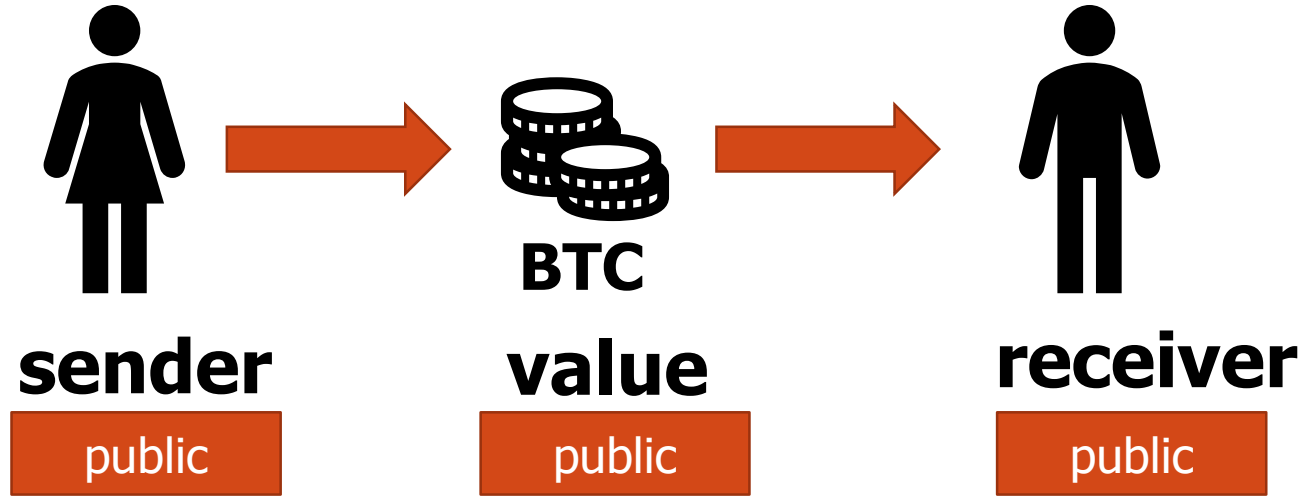
Background

Bitcoin is ~~Anonymous~~ pseudonymous!



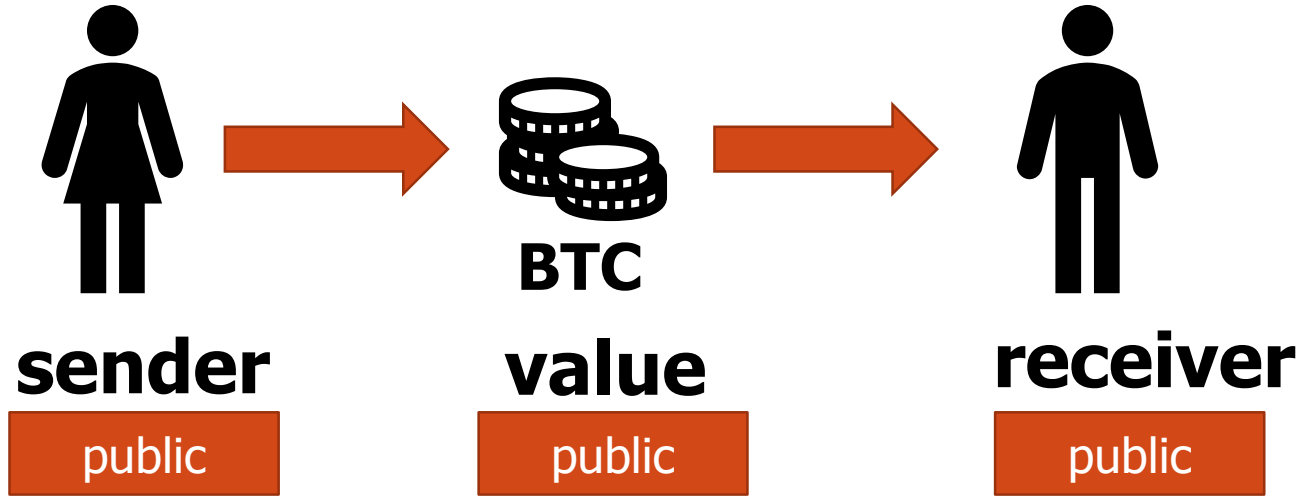
Background

Bitcoin is ~~Anonymous~~ pseudonymous!



Background

Bitcoin is ~~Anonymous~~ pseudonymous!

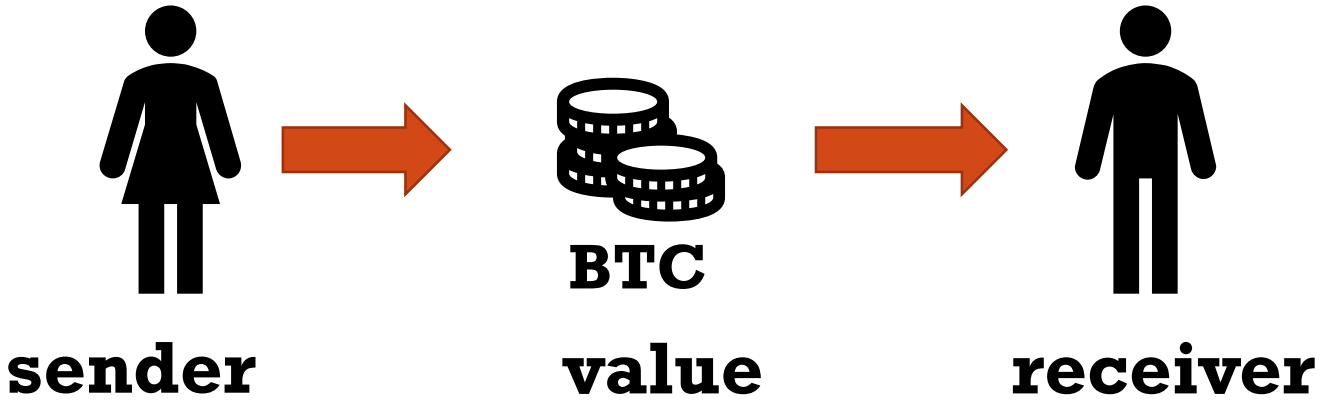


Untraceability:* Given a transaction input (output), the real output (input) should be anonymous among a set of other outputs (inputs).

* Kumar, A., Fischer, C., Tople, S., & Saxena, P. (2017, September). A traceability analysis of monero's blockchain. In *European Symposium on Research in Computer Security* (pp. 153-173). Springer, Cham.

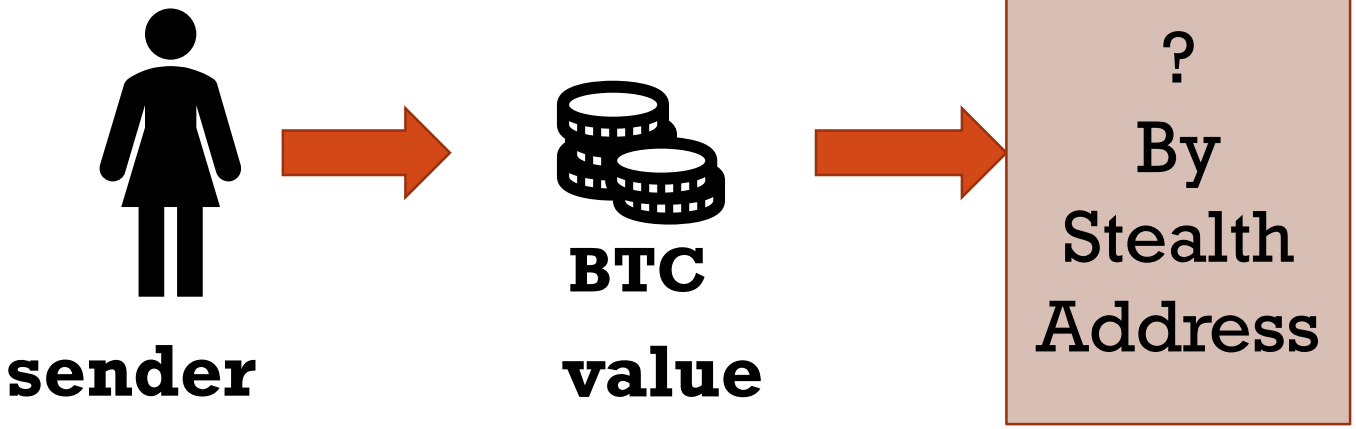
Background

MONERO gains privacy preservation with



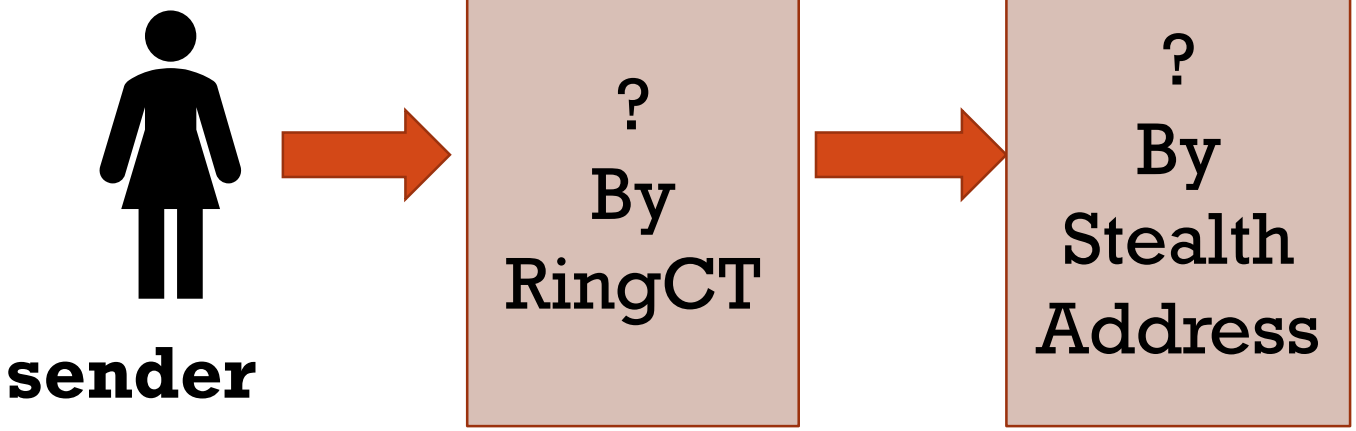
Background

MONERO gains privacy preservation with



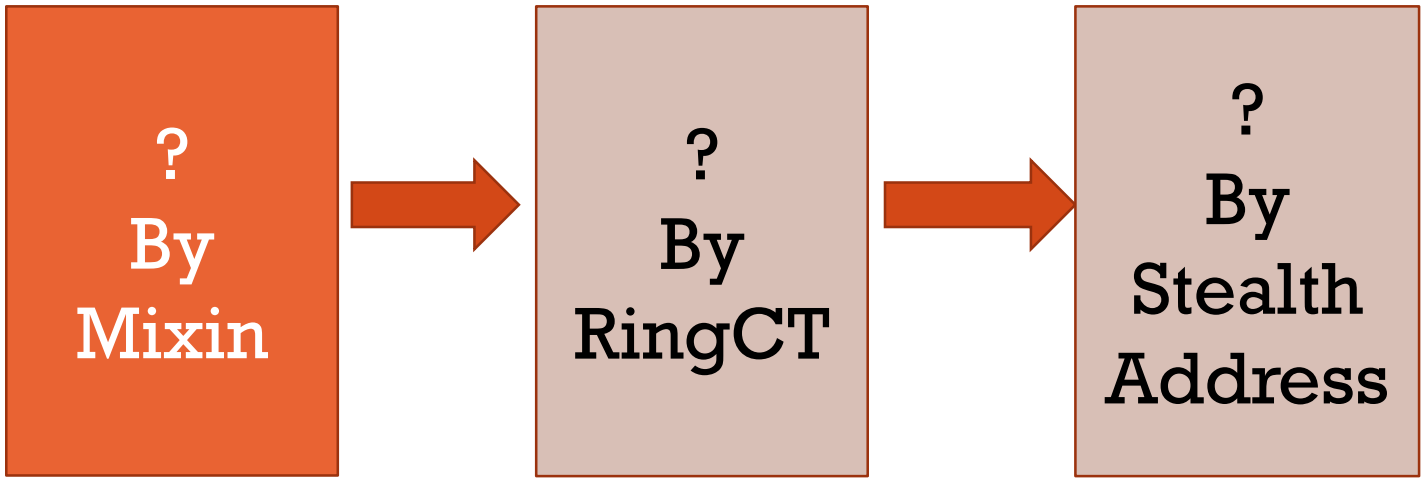
Background

MONERO gains privacy preservation with



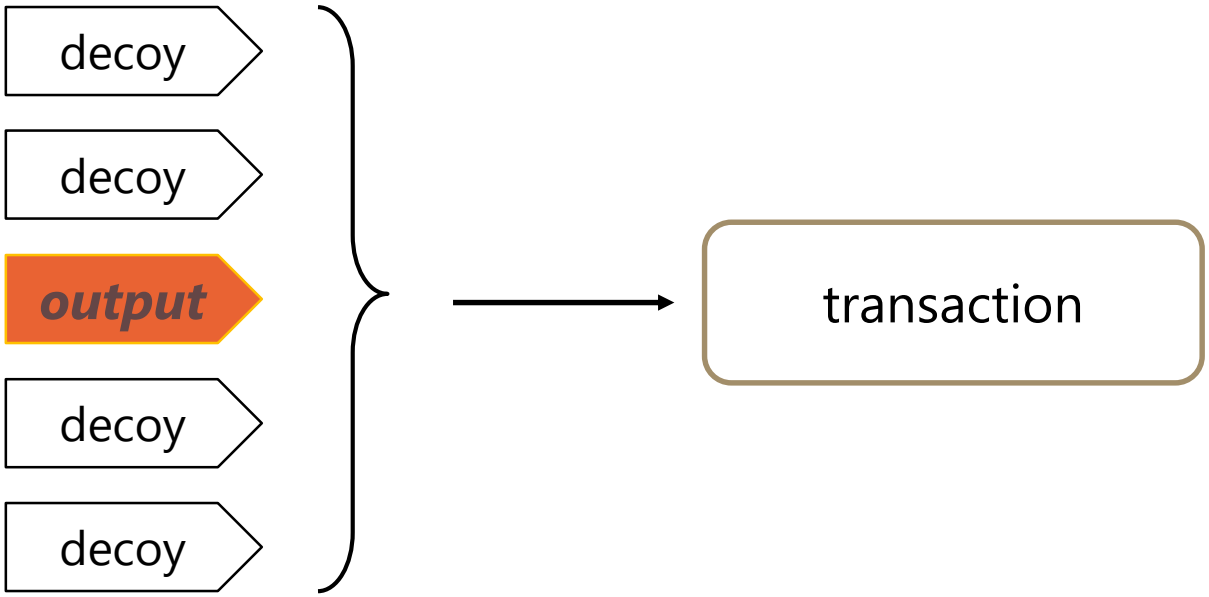
Background

MONERO gains privacy preservation with



Mixin: applied by Monero to obfuscate the real transaction input from previous transaction outputs.

Background



INPUT (with mixin = 4)

The transaction input in referenced by totally 5 previous outputs, **making it confusing** to know which is the real input.

Motivation

- Recent works have empirically shown the probability of **traceability** in Monero blockchain, mainly .
- Analyze the reason for this privacy-breaking by mixin.
- Design to improve the mix choosing (sampling) algorithm

Evaluation index: GE

- **Guessing entropy** is commonly used as a measure of password strength*. A transaction input with size of m, has the guessing entropy

$$GE = \sum_{0 \le i \le m} i \times p_i$$

where p = p0, p1, ... ,pm are probabilities, sorted highest to lowest, that a referenced output is the real spend of a transaction input.

- In the context of Untraceability, guessing entropy is the expected number of guesses before guessing the spent output.

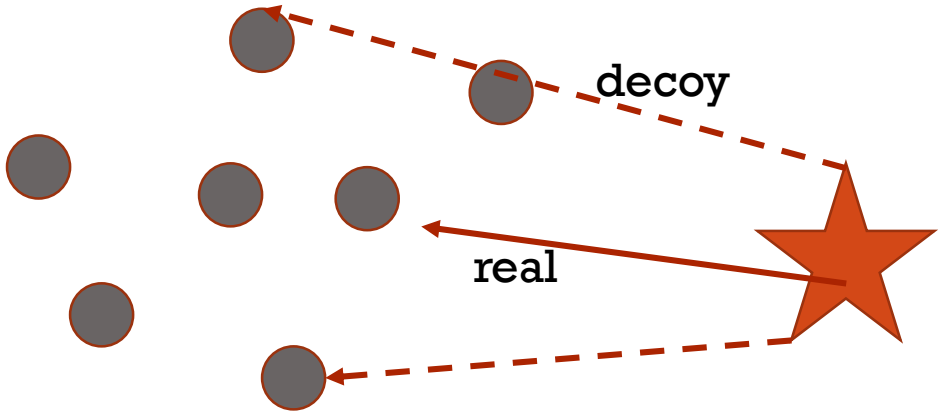
*Miller, A., Moeser, M., Lee, K., & Narayanan, A. (2017). An empirical analysis of linkability in the monero blockchain.

Monte Carlo Simulation

Dataset: https://senseable2015-6.mit.edu/bitcoin/

Up to 6GB, including Bitcoin transactions from Block 446000 – 500000
(Approximately Jan 2017 – Feb. 2018)

Uniform sampling



- Select the mixin samples **randomly** (with equal probability) in the previous network.
- Applied by Monero early.

Uniform sampling

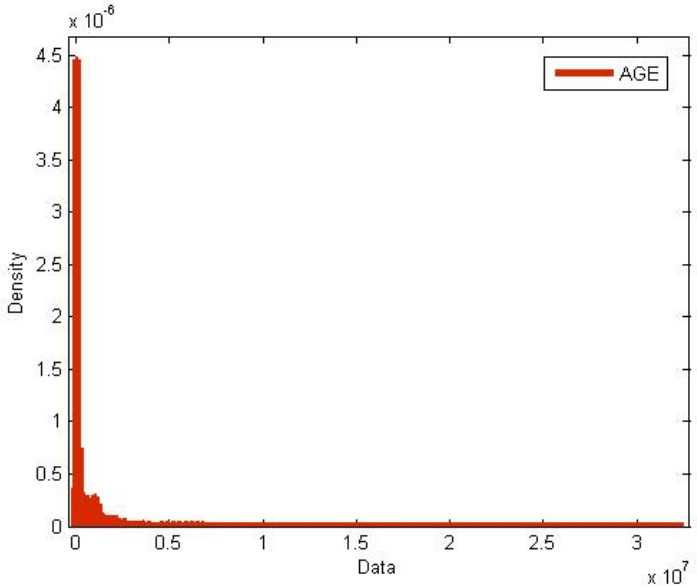


Fig. age distribution of the observed dataset.

- Bitcoin transaction output age distribution shows **imbalance**, influencing the effectiveness of random sampling.
- (age = in how much time a transaction output is spent later)
- One can always **guess the "newest"** as the transaction real input with several mixin ones.

Distribution fitted sampling

If we know the age distribution, we can improve the sampling method by making **sampling distribution = real transaction age distribution.**

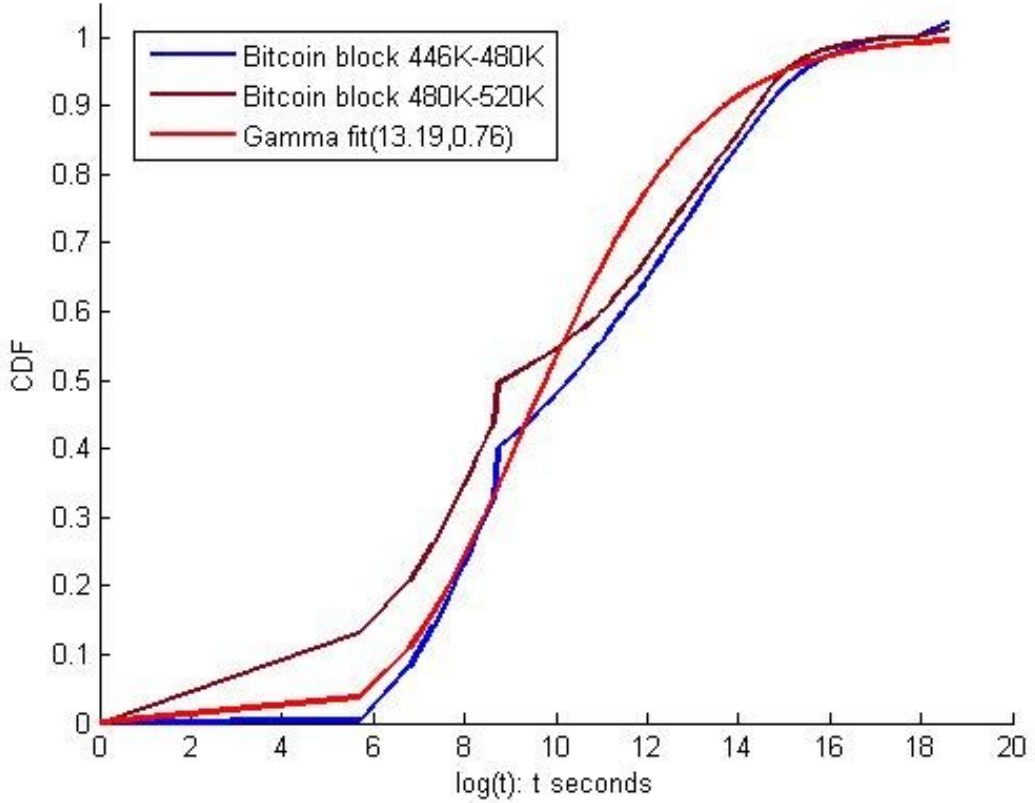


Fig. Gamma fitting of age-CDF with shape=13.19, rate=0.76

Random vs. Distribution fitted sampling

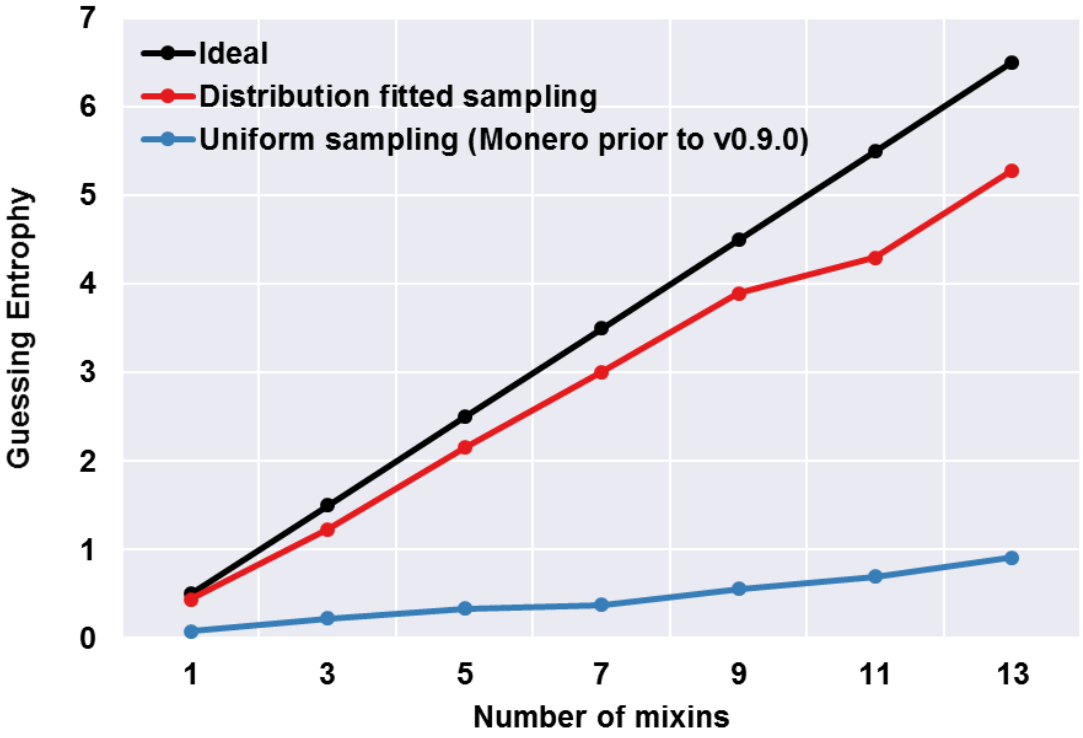


Fig. GE of different mixins (50k simulations)

We gain remarkable improvement by changing the uniform sampling to distribution fitted sampling.

Problem: the real distribution may be different from sampling distribution

What if real distri. ≠ sampling distri.

For a given samling list X, each xi has the probability of

$$P(x_i|X) = \frac{P(x_i)P(X|x_i)}{P(X)}$$

$$= \frac{D_S(x_i)\prod_{0\le j\neq i\le m} D_M(x_j)}{\sum_{0\le p\le m}\left(D_S(x_p)\prod_{0\le q\neq p\le m} D_M(x_q)\right)}$$

, where Ds is the real spent distribution, and Dm the mixing sampling distribution. By letting r_k = Ds(x_k)/Dm(x_k), we get GE

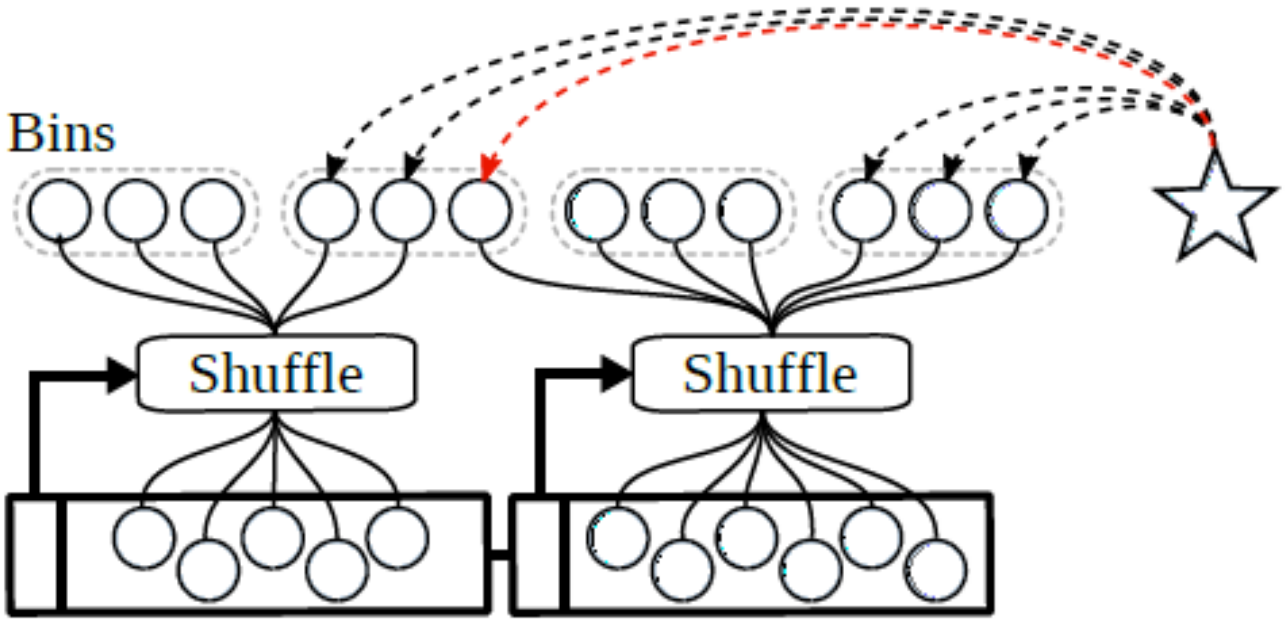
$$\mathrm{Ge} = \sum_{0\le k\le m} k\cdot\frac{r_k}{\sum_{0\le p\le m}(r_p)}, \quad r_{min} = \min_{\forall x}\left(\frac{D_S(x)}{D_M(x)}\right)$$ = 1- e

Different e values makes the Ge different from previous conclusion.

Attackers can make traceability happen!*

[1] Mackenzie, A., Noether, S., & Team, M. C. (2015). Improving Obfuscation in the CryptoNote Protocol

Binned sampling algorithm



To get a mixin list, we do three steps:*
(1) Shuffle transactions in each block, and form bins with fixed size.
(2) Choose several bins from the network with gamma distribution
(3) Add all th transactions in the bins to the mixin list

Miller, A., Möser, M., Lee, K., & Narayanan, A. (2017). An empirical analysis of linkability in the Monero blockchain. *arXiv preprint arXiv:*

Binned sampling algorithm

		GE					
		Binsize	e = 0%	25%	50%	75%	100%
5	mixins	1	6	**5.43**	4.33	2.43	1
5	mixins	2	6	5.18	4	2.67	2
5	mixins	3	6	5.16	4.2	**3.35**	**3**
7	mixins	1	8	**7.38**	6.09	3.43	1
7	mixins	2	8	7.02	5.43	3.26	2
7	mixins	4	8	6.88	5.6	**4.47**	**4**
8	mixins	1	9	**8.36**	**7**	4	1
8	mixins	3	9	7.76	6	4	**3**

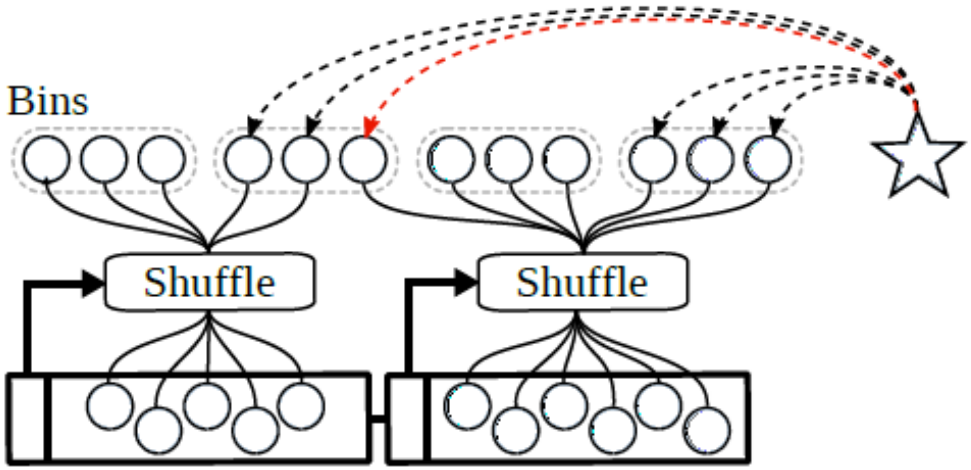
Table. Theoretical GE value of different e, bin size.

- A binned sampling method can **relieve** the negative influence of difference of mixing and real distribution.

Binned sampling algorithm

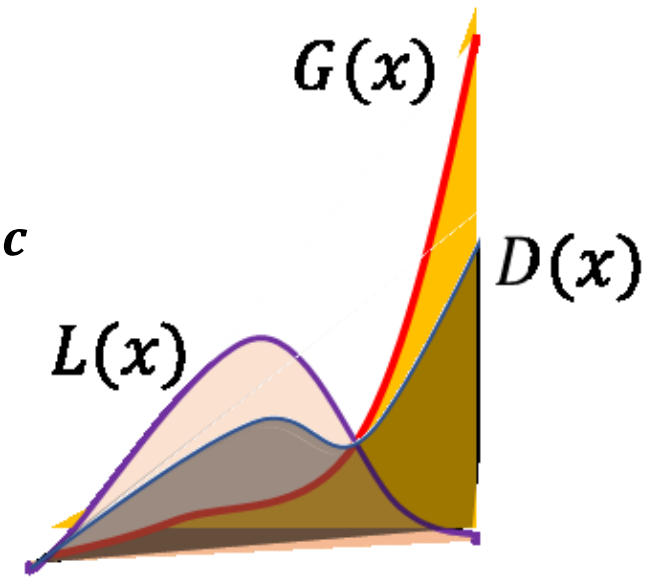
- Binned sampling is used to relieve distribution attack. If an attacker finds abnormal sampling results (and if he knows recently the transaction spending distribution could not be like this), he can know with high probability of which is the real input.

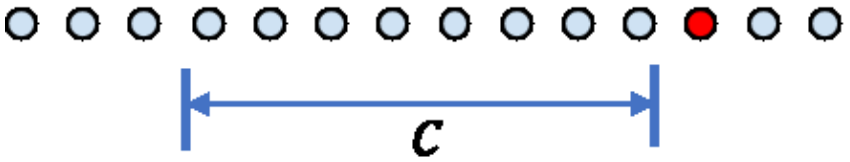
- With bins, even if the attacker knows where the real input is, he will get a "bin" of **several choices instead of only one**, preventing him from knowing which is the real input.



Binned sampling with sliding window

- A more general model to deal with the bin sampling of global gamma distribution $G(x)$ **and local distribution** $L(x)$
- **Sliding window**: calculate the local distribution $L(x)$
  - **Static sliding window**: a fixed size c
  - **Dynamic sliding window**: an unfixed size $c(t)$
- $D(x) = aG(x) + bL(x) + \varphi(x)$
- A larger amount of calculation





Binned sampling with sliding window

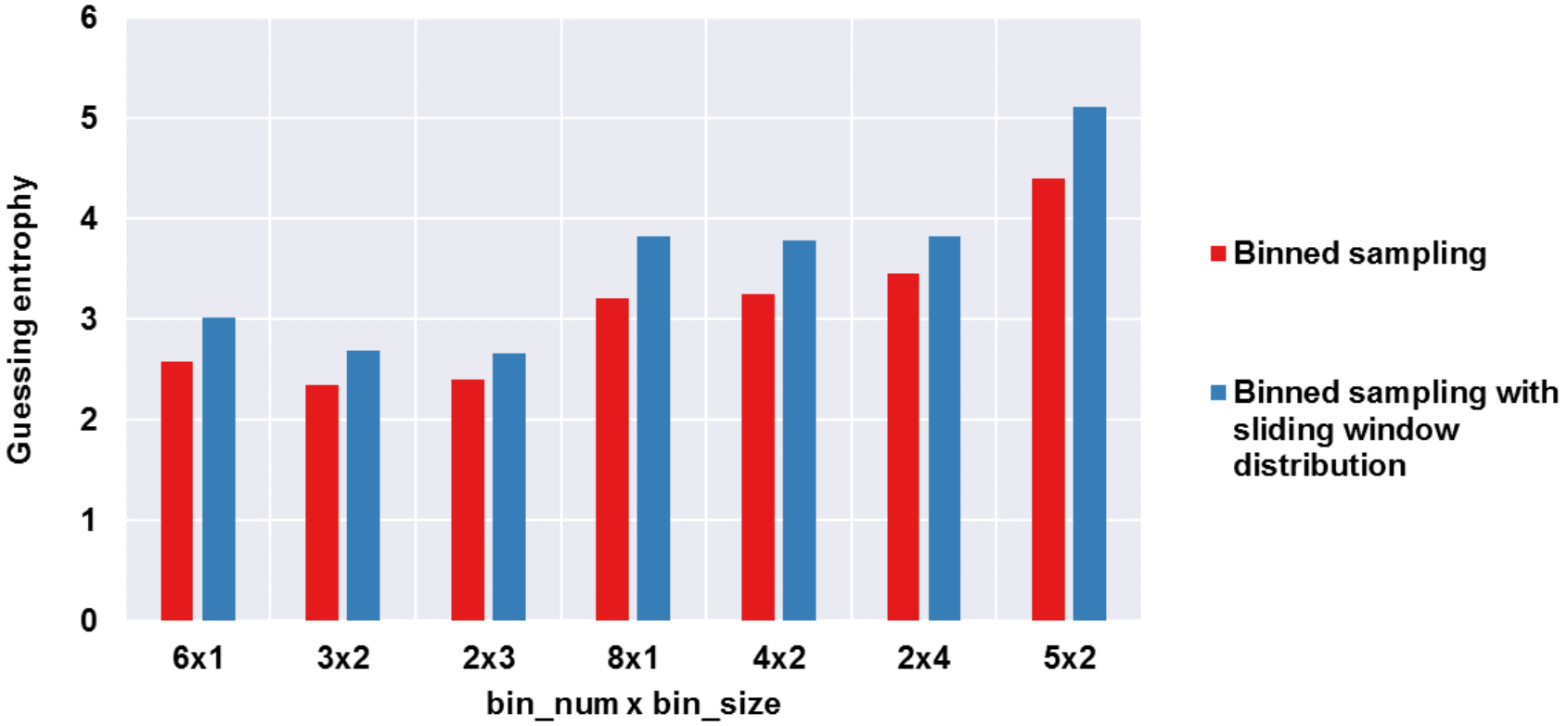


Fig. The GE value (100k simulations) with sliding windows is **higher (15 % rise),** using different bin size and bin number.

Conclusions

- The monero mixin sampling may not be effective, if the sampling **fails to meet** the real spending age distribution.
- We can get high improvement on untraceability, if we know exactly the spending age distribution. But this is not always the case.
- A binned sampling **to relieve the negative effects** with different sampling and spending distribution.
- **Sliding windows** help to better fit the local, recent spending distribution, improving the untraceability.

Strength and weakness

- Strength

Targeting at the mixin algorithm.

Simulations to gain untraceability value.

A better mixin algorithm with 15% raise in untraceability.

- Weakness

Bitcoin data is not the same as Monero.

Simulations are not enough.

More theoretical analysis is needed.

Improving the Mixin Sampling Algorithm for Better Untraceability in Monero Blockchain

Thanks!

Yujie Pan
2018.6.1