2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) | 978-1-6654-3578-9/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICBC51069.2021.9461130

New Attacks on the Untraceability of Transactions in CryptoNote-Style Blockchains

Sina Aeeneh
Department of ECSE, Monash University
Melbourne, Australia
mohammadsina.aeeneh@moansh.edu

João Otávio Chervinski
Faculty of Information Technology, Monash University
Melbourne, Australia
joao.massarichervinski@monash.edu

Jiangshan Yu
Faculty of Information Technology, Monash University
Melbourne, Australia
jiangshan.yu@monash.edu

Nikola Zlatanov
Department of ECSE, Monash University
Melbourne, Australia
nikola.zlatanov@monash.edu

***Abstract*—CryptoNote is a privacy-focused blockchain protocol. Currently, more than 10 different cryptocurrencies including Monero, a popular cryptocurrency with a 1-year average market cap of 2 billion USD, have been developed based on the CryptoNote's protocol. CryptoNote obscures the connection between inputs and outputs of a transaction by adding decoy inputs. In this paper, we introduce probabilistic attacks to the untraceability of CryptoNote which substantially degrades the level of privacy of CryptoNote-style blockchains. We analyze the effectiveness of the proposed attacks and derive their error rates.**

***Index Terms*—blockchain, CryptoNote, privacy, Monero, information theory.**

I. Introduction

BLOCKCHAIN is a technology that can provide consensus among users of a distributed network. The state of the system is recorded as a chain of blocks and is distributed throughout the network so that all users can update their local information about the current state of the system and reach an agreement. The state of the system is regularly updated by adding new blocks to the chain. Each block stores a number of new transactions which contains the record of all changes that have been made in the system since the previous block.

Preserving the privacy of the users is of great importance in computer networks and becomes more crucial when dealing with public, open and decentralized systems, i.e., blockchains. The lack of privacy in a blockchain-based system can disclose users' private information to everyone in the network rather than a centralized authority only.

CryptoNote is a privacy-focused blockchain protocol that addresses untraceability of transactions [1]. A transaction is untraceable if the connections between inputs and outputs of the transaction are unclear to the a blockchain observant [2]. CryptoNote transactions may have several inputs and outputs. A user issuing a transaction has the option to obscure each input within a set of l inputs, where one is the real input and the rest are decoys. The issuer of the transaction, then, introduces those sets as inputs without revealing real inputs that are transferred as a result of the transaction. Using the cryptographic features of *one time ring signatures*, double spending is prevented and the network will place the transaction in the blockchain, only if the issuer has the right to spend at least one element of each set of the potential inputs it provides in the transaction [3]. In this paper, we propose two attacks that significantly weaken the untraceability feature of CryptoNote-style blockchains.

The rest of this paper is organised as follows. In Section II we explain the untraceability feature of CryptoNote in more details. Next, we propose two attacks that significantly weakens this feature in Sections III and IV. Finally, Section V concludes the paper.

II. Untraceability of CryptoNote

CryptoNote achieves untraceability using the ring signature protocol [3]. If Alice wants to transfer some cryptocurrency to Bob, she issues a new transaction and uses a subset of the transaction outputs (TXOs) she possesses their respective private-keys to transfer all or a portion of her cryptocurrency. In order to make her transaction untraceable, for each one of the TXOs that she wants to transfer to Bob, Alice uses the blockchain's database and selects $l-1$ decoy TXOs. Next, she puts each true TXO next to $l-1$ decoys and creates a ring signature. Let $\mathcal{G}i = \{g_1, g_2, ..., g_l\}$ be such a group where we call the elements of the set as mixins.

Exploiting the cryptographic properties of the one-time ring signature protocol, Alice uses the private-keys she owns and signs the transaction while announcing the elements of $\mathcal{G}i$'s as the potential spending of the transaction. By signing the transaction, Alice generates new TXOs and transfers the right for spending them to Bob. Such a transaction is valid and will be accepted by miners if the elements in $\mathcal{G}i$'s exist on the blockchain and Alice has the right to sign the transaction, i.e.,

We would like to show our gratitude to Ms. Niusha FarajMohammadi for her helpful comments and assistance in revising the paper.

978-1-6654-3578-9/21/$31.00 ©2021 IEEE

she has the respective private-key for at least one element in each $\mathcal{G}i$ and she has never spent it before.

Due to the cryptographic properties of one-time ring signature, the miners in the network can verify that the transaction is valid, while they cannot precisely identify which elements of $\mathcal{G}i$'s is spent in that transaction. As a result, each input of a transaction remain anonymous within a set of l potential inputs. Although a larger ring size provides a better untraceability, it also increases the cost of having the transaction validated, since the transaction fee is proportional to the size of a transaction, which increases by the ring size.

Although the CryptoNote protocol broadly suggests the mixins to be randomly selected from the previous blocks on the blockchain, further studies showed that the way users select the decoys of their transactions may affect their privacy as well as others' [4]–[6]. For example, an adversary might analyze the record of the previous transactions on the blockchain and identify some of the elements in $\mathcal{G}i$ as decoys. Alternatively, he might exploit the record of the previous transactions on the blockchain and obtain probabilistic hypotheses about the truly spent TXOs in a transaction. Therefore, Alice has to be careful in selecting the decoys so that an adversary cannot destroy or weaken the untraceability promised by the protocol.

Let's define the age of a mixin as the difference between the block height in which it is used as a mixin and the block height in which it is produced, i.e., mined as an output of a transaction. In other words, if a TXO is recorded as N'th transaction on the blockchain and appears in the mixin set of the M'th transaction on the blockchain, then, we define the age of that mixin as $M - N$. [1] CryptoNote proposal suggests selecting decoys randomly without providing further explanation [1], however, its reference implementation utilizes a uniform distribution for selecting decoys from all previous outputs on the blockchain [4].

Authors in [4] showed that the age of a TXO at the time of being transferred by the owner follows a distribution similar to the Gamma distribution. Therefore, if the age of the decoys does not follow the same distribution, a simple analysis on the age of the inputs of each TXO can result in guessing the truly spent TXO with a probability higher than $\frac{1}{l}$.

In another study, the authors argued that, in the majority of cases, the newest TXO in the mixin set is the truly spent one [2]. They verified their heuristic by simulation and showed that 99.5% of the inputs of transactions can be traced using this simple attack. Since April 2015 Monero developers have switched to a triangular distribution for selecting previous TXOs as decoys. However, using the same attack still 96% of the transactions can be traced [2].

Monero developers also used a uniform mixin selection in their initial implementation of CryptoNote, however, they changed the mixin selection algorithm several times afterward and from version 0.13.0 onward, Monero switched to the Gamma distribution for selecting decoys as suggested by [4].

Using a better distribution, i.e., a distribution closer to the distribution of the age of the truly spent TXOs can make the attack proposed in [2] less effective. However, since the users' behavior changes over time [7], it is hard to constantly find and follow the distribution of the age of the truly spent TXOs and select the decoys of transactions using the exact same age distribution.

III. Attack 1; Ignored TXO

In this Section, we introduce and analyze our first attack. In this attack we argue that since the users are utilizing a probabilistic mixin selection algorithm, there is a chance that some TXOs are accidentally ignored by other users and are not used as a mixin of any transaction but their real spending ones. This may happen even if the sender and receiver of a transaction are following the protocol with honesty.

For simplicity of the explanation, we assume that the network only approves transactions with the ring size equal to l. We assume that all users but a negligible portion are using wallet source codes that employ the same distribution as the official protocol for selection of the decoys. While this assumption is not necessary for implementation of Attack 1, it is needed for the analysis.

Let $P_t(n)$ denote the probability that a new TXO, X_N, will be transferred when exactly n new TXOs are recorded on the blockchain after X_N. Therefore, the probability of a TXO remaining unspent after n TXOs, denoted by $\epsilon(n)$, is given by

$$\epsilon(n) = 1 - \sum_{i=1}^{n} P_t(n). \tag{1}$$

From (1) we can see that $\epsilon(n)$ is a descending function of n. The value of $\epsilon(n)$ depends on the behaviour of the users. For instance, in a blockchain like Bitcoin where the users hold the cryptocurrency as a long-term investment, $\epsilon(n)$ could be as large as 0.55 for $n = 10^8$, that is the number of transactions in one year [8]. However, if a cryptocurrency is mainly used as a medium of exchange where users frequently exchange the currency for goods and services, then $\epsilon(n)$ could be very close to zero for a large enough n. In the rest of this paper we assume that $\epsilon(n)$ has a negligible value for large n's in order to make the equations more tractable, however, the proposed attacks can be used for any value of $\epsilon(n)$.

In order to obtain $\epsilon(n)$ in (1) we need to know the distribution of the age of the truly spent TXOs. Proposition 1 exploits the distribution of the age of TXOs when they appear as an input of a transaction as well as the distribution used in the mixin selection algorithm in order to extract the distribution of the truly spent TXOs, $P_t(.)$.

Proposition 1: The probability mass function (PMF) of the truly spent TXOs is given by

$$P_t(a) = lP_b(a) - (l-1)P_f(a), \tag{2}$$

where $P_b(.)$ is the distribution of the age of TXOs when they are used as elements of ring signatures derived from the blockchain and $P_f(.)$ is the distribution that the mixin selection algorithm uses for selecting decoys.

[1] Alternatively, one can define the age of a mixin as the difference between block heights instead of the transaction heights, however, we stick to the former definition.

Proof: We derive the probability of a TXO being observed on the blockchain with age a as

$$\begin{aligned} P_b(a) =& \Big(Pr(\text{fake}) Pr(a|\text{fake}) + Pr(\text{true}) Pr(a|\text{true}) \Big) \\ P_b(a) =& \frac{l-1}{l} P_f(a) + \frac{1}{l} P_t(a). \end{aligned} \tag{3}$$

Therefore, we have

$$P_t(a) = l P_b(a) - (l-1) P_f(a). \tag{4}$$

□

Let $P_0(K)$ be the probability of a specific TXO remains ignored for K transactions, i.e., not utilized as a decoy in the next K transactions after being issued. We obtain $P_0(K)$ as

$$P_0(K) = \prod_{k=1}^{K} \Big(1 - P_f(k)\Big)^{(l-1)}. \tag{5}$$

Let X_i be the i'th TXO on the blockchain. For each X_N, we scan the mixin sets of all transactions with outputs in $\{X_{N+1}, ..., X_{N+K}\}$. If X_N appears in the mixin set of only one transaction, then X_N is very likely the truly spent one in that transaction. If K is large enough, then the error rate decreases and our inference will have a higher precision since users usually spend their TXOs after a limited time. Proposition 2 gives the precision of such an inference.

Proposition 2: Let X_N be an output of a transaction which is used as a mixin exactly k_1 transactions later, for the first time. If X_N is not used as a mixin of any other transaction for at least k_2 transactions after the first time, then the probability that the first time that it appeared on the blockchain is its true spending, denoted as $P(A|B_{k_1,k_2})$, is given by

$$P(A|B_{k_1,k_2}) = \frac{P_t(k_1)}{P_t(k_1) + \epsilon(k_1+k_2) \frac{1 - \left(1 - P_f(k_1)\right)^{(l-1)}}{\left(1 - P_f(k_1)\right)^{(l-1)}}}, \tag{6}$$

where events A and B_{k_1,k_2} are defined as

A is the event in which X_N is spent in the k_1'th transactions after its issuance.

$\mathbf{B_{k_1,k_2}}$ is the event in which X_N is not used in any of the $k_1 + k_2$ transactions after its issuance except for the k_1'th.

Proof: We derive $P(A|B_{k_1,k_2})$ as

$$\begin{aligned} P(A|B_{k_1,k_2}) =& \frac{P(B_{k_1,k_2}|A)P(A)}{P(B_{k_1,k_2}|A)P(A) + P(B_{k_1,k_2} \cap \overline{A})} \\ =& \frac{P_0(k_1+k_2)P_t(k_1)}{P_0(k_1+k_2)P_t(k_1) + \epsilon(k_1+k_2)P_0(k_1+k_2) \frac{1-\left(1-P_f(k_1)\right)^{(l-1)}}{\left(1-P_f(k_1)\right)^{(l-1)}}} \\ =& \frac{P_t(k_1)}{P_t(k_1) + \epsilon(k_1+k_2) \frac{1-\left(1-P_f(k_1)\right)^{(l-1)}}{\left(1-P_f(k_1)\right)^{(l-1)}}}. \end{aligned} \tag{7}$$

□

Let K be the smallest number for which $\epsilon(K)$ is negligible compared to $P_t(k_1)$ for all k_1's smaller than K. Then, for all k_1 and k_2's where $k_1 + k_2 > K$ the probability in (6) is very close to 1 and Attack 1 will have a very high precision. Proposition 3 indicates a lower bound on the percentage of transactions being traced with a precision close to 1 using Attack 1.

Proposition 3: Assuming that $\epsilon(n)$ diminishes as n grows, the probability of a randomly selected transaction being traced almost surly using Attack 1, denoted by $Pr(\text{Attack 1})$, is given by

$$Pr(\text{Attack 1}) \approx P_0(K+1). \tag{8}$$

Proof: The proof is provided in Appendix A.

In those cases where X_N shows up as a mixin of m transactions where $1 < m < l$ we can use a weaker version of Attack 1 to reduce the size of the anonymity set from l to m with a high precision. To this end, for each TXO like X_N, we scan the next transactions with outputs in $\{X_{N+1}, X_{N+2}, ..., X_{N+K}\}$ and search for those which use X_N as a mixin. Recalling from (1), we know that X_N is spent in one of those transactions with probability $1 - \epsilon(K)$. Let $f(X_N, K)$ denote the number of times X_N appears in the mixin set of the next K outputs. If K is large enough $\epsilon(K) \approx 0$, and the ambiguity about the true spending of X_N will be between $f(X_N, K)$ transactions. The probability of such an event, i.e., having m-anonymity about a TXO, is given by

$$Pr\big(f(X_N, K) = m\big) \approx P_{rec}(m-1, K), \tag{9}$$

where $P_{rec}(.,.)$ is defined as

$$P_{rec}(n-1, K_n) = \sum_{K_{n-1}=N+n}^{K_n} P_f(K_{n-1}) P_{rec}(n-2, K_{n-1}). \tag{10}$$

IV. Attack 2: MAP Decoder

The attack proposed in this section is similar to the one introduced in [2]. However, what we propose is optimal and can be implemented in scenarios where the attack introduced in [2] does not work.

Recalling $P_t(.)$ as the distribution of the age of truly spent TXOs and $P_f(.)$ as the distribution from which users select their decoys, if $P_t(.)$ and $P_f(.)$ do not match, we can guess the real spend of each mixin set with a probability higher than $\frac{1}{l}$, therefore, weakening the untraceability promises of CryptoNote. Although using better distributions for mixin selection can resolve the attack proposed in [2], they cannot eliminate the information obtained from the age of mixins completely unless $P_f(.)$ exactly matches with $P_t(.)$. Since the behaviour of the users changes over time, it is very difficult to implement a mixin selection algorithm that perfectly follows the true spending distribution of TXOs.

Let $\vec{\mathcal{G}} = (g_1, g_2, ..., g_l)$ denote the vector of mixins in a ring signature, we denote the vector of the age of mixins as $\vec{Y} = (y_1, y_2, ..., y_l)$, where y_i denotes the age of g_i. Attack

2 tries to guess the real transaction input among $\vec{\mathcal{G}}$ with the minimum probability of error.

Using the vector of the ages of the TXOs in $\vec{\mathcal{G}}$, we declare g_i as the true spending if

$$i = \underset{i\in\{1,2,\ldots,l\}}{\operatorname{argmax}} \left\{\frac{P_t(y_i)}{P_f(y_i)}\right\}. \tag{11}$$

Proposition 4: The decision rule in (11) is the optimal decision rule in terms of error probability when using $\vec{Y}$ as the input information.

Proof: Starting from the Maximum A Posteriori (MAP) estimation, we have

$$\begin{aligned}\hat{g} = g_j \iff j &= \underset{i\in\{1,2,\ldots,l\}}{\operatorname{argmax}} Pr\Big(g_i \text{ is true} \mid \vec{Y}\Big)\\ &= \underset{i\in\{1,2,\ldots,l\}}{\operatorname{argmax}} Pr\Big(\vec{Y} \mid g_i \text{ is true}\Big)Pr\Big(g_i \text{ is true}\Big)\\ &= \underset{i\in\{1,2,\ldots,l\}}{\operatorname{argmax}} \Big\{P_t(y_i)\prod_{j=1,j\neq i}^{l} P_f(y_j)\Big\}\\ &= \underset{i\in\{1,2,\ldots,l\}}{\operatorname{argmax}} \Big\{\frac{P_t(y_i)}{P_f(y_i)}\Big\}, \end{aligned} \tag{12}$$

where $\hat{g}$ is the optimal estimation which minimizes the probability of error. In the last equality we assume that the true TXI is in one of the l positions in $\vec{\mathcal{G}}$ with equal probability. □

Proposition 5: The error probability of attack 2 is given by

$$Pr(\text{error}) = 1 - \sum_{y_1=1}^{\infty} P_t(y_1)\Big(\sum_{y\in\mathcal{S}(y_1)} P_f(y)\Big)^{l-1}, \tag{13}$$

where $S(y_1)$ denotes the sub-domain of $P_f(y)$ where $\frac{P_t(y_1)}{P_f(y_1)} > \frac{P_t(y_i)}{P_f(y_i)}$ holds.

Proof:

$$\begin{aligned} Pr(\text{error}) =& 1 - Pr(\text{correct})\\ =&1 - \sum_{i=1}^{l} Pr(\text{correct} \mid g_i \text{ is true})Pr(g_i \text{ is true})\\ =&1 - Pr(\text{correct} \mid g_1 \text{ is true})\\ =&1 - \sum_{y_1=1}^{\infty} P_t(y_1)\prod_{i=2}^{l} Pr\Big(\frac{P_t(y_1)}{P_f(y_1)} > \frac{P_t(y_i)}{P_f(y_i)}\Big)\\ =&1 - \sum_{y_1=1}^{\infty} P_t(y_1)\Big(\sum_{y\in\mathcal{S}(y_1)} P_f(y)\Big)^{l-1}. \end{aligned} \tag{14}$$

□

Extending attack 2, we can extract the probability of each element of $\vec{\mathcal{G}}$ be the true spending as

$$Pr(g_i \text{ is true}) = \frac{\frac{P_t(y_i)}{P_f(y_i)}}{\sum_{j=1}^{l} \frac{P_t(y_j)}{P_f(y_j)}}. \tag{15}$$

It is desired for ring signatures to prevent leaking information from the age of mixins. Therefore, ideal ring signatures have to have equal probabilities in (11), otherwise, an adversary may reduce the ambiguity about the true spending.

Let random variable $X \in \{1, 2, ..., l\}$ denote the index of the true spent TXO in $\vec{\mathcal{G}}$. Proposition 6 uses Shannon information [9] in order to derive the mutual information between X and $\vec{Y}$ as a metric for measuring the amount of information leaked from the age of mixins.

Proposition 6: The information about the true spending of a mixin set, obtained from $\vec{Y}$ is given by

$$\begin{aligned} I(X;\vec{Y}) =&H(\vec{Y}) - H(\vec{Y}|X)\\ &= -\Big(\sum_{i=1}^{l}\sum_{y_1^i} P_{Y_1^i}(y_1^i)\log_2\frac{P_{Y_1^i}(y_1^i)}{P_{Y_1^{i-1}}(y_1^{i-1})}\Big)\\ &- \Big((l-1)H(f) + H(t)\Big), \end{aligned} \tag{16}$$

where $H(t)$ is the entropy of distribution $P_t(.)$, $H(f)$ is the entropy of distribution $P_f(.)$, and $P(y_1^i)$ is given by

$$P_{Y_1^i}(y_1^i) = \frac{1}{l}\sum_{x=1}^{i} P_t(y_x)\prod_{j=1,j\neq x}^{i} P_f(y_j) + \frac{l-i}{l}\prod_{j=1}^{i} P_f(y_j). \tag{17}$$

Proof: The proof is provided in Appendix B.

V. Conclusion

Two different attacks on untraceability of CryptoNote have been proposed where the first attack exploits the possibility that some TXOs have not been chosen as decoys, simply because the decoy selection algorithm is random. Probability of error and the proportion of transactions affected by different variations of this attack have been derived. The second attack exploits the distance between the age distribution of the truly spent TXOs and the age distribution of decoys in order to decode mixin sets and infer the TXO that has been transferred in each set. The probability of error of the second attack and the maximum information obtained from observing each mixin set is derived. We showed that if the distribution used for selecting decoy TXOs are known, then a malicious party can significantly weaken the privacy features of CryptoNote by tracing back the transactions. However, our studies on Monero, a CryptoNote-based blockchain, show that users are using different source codes each having a different algorithms for mixin selection. Fortunately, since there is no distribution that a strong majority of users are using for mixin selection, an attacker cannot derive the distribution of truly spent transactions and arrange the second attack on Monero. However, the first attack is still applicable, though the analysis of the effectiveness of the attack in this situation is challenging to obtain and requires further research.

APPENDIX A
PROOF OF PROPOSITION 3

Attack 1 is successful with a precision close to 1 if event B_{k_1,k_2} holds for different k_1 and k_2's as long as $k_1 + k_2 > K$ holds. Therefore, we have

$$\begin{aligned}
Pr(\text{Attack 1}) &= Pr(\bigcup_{\substack{k_1,k_2\\ k_1+k_2>K}} B_{k_1,k_2}) \\
&= Pr(\bigcup_{\substack{k_1,k_2\\ k_1+k_2=K+1}} B_{k_1,k_2}) \\
&= \sum_{\substack{k_1,k_2\\ k_1+k_2=K+1}} P(B_{k_1,k_2}|A)P(A) + P(B_{k_1,k_2} \cap \overline{A}) \\
&= \sum_{k_1=1}^{K+1} P_0(K+1)P_t(k_1) \\
&+\epsilon(K+1) \sum_{k_1=1}^{K+1} P_0(K+1) \frac{1-\Big(1-P_f(k_1)\Big)^{(l-1)}}{\Big(1-P_f(k_1)\Big)^{(l-1)}} \\
&\approx P_0(K+1) \sum_{k_1=1}^{K+1} P_t(k_1) \\
&= P_0(K+1)\Big(1-\epsilon(K+1)\Big) \\
&\approx P_0(K+1), \qquad (18)
\end{aligned}$$

where we assumed $\epsilon(K+1) \approx 0$. □

APPENDIX B
PROOF OF PROPOSITION 6

Starting from the chain rule for the entropy, we have

$$\begin{aligned}
I(X;\vec{Y}) =& H(\vec{Y}) - H(\vec{Y}|X) \\
=& \sum_{i=1}^{l} H(Y_i|Y_1^{i-1}) - \sum_{i=1}^{l} H(Y_i|X) \\
=& -\Big(\sum_{i=1}^{l}\sum_{y_1^i} P_{Y_1^i}(y_1^i)\log_2 \frac{P_{Y_1^i}(y_1^i)}{P_{Y_1^{i-1}}(y_1^{i-1})}\Big) \\
& -\Big((l-1)H(f) + H(t)\Big), \qquad (19)
\end{aligned}$$

where $P_{Y_1^i}(y_1^i)$ denotes the probability of $y_1^i = (y_1, y_2, ..., y_i)$ being the realization of the random vector $Y_1^i = (Y_1, Y_2, ..., Y_i)$, and is driven as

$$\begin{aligned}
P_{Y_1^i}(y_1^i) &= \sum_{x=1}^{i} P_{Y_1^i}(y_1^i|X=x)P_X(x) + P_{Y_1^i}(y_1^i|X>i) \\
&= \frac{1}{l}\sum_{x=1}^{i} P_t(y_x) \prod_{j=1,j\neq x}^{i} P_f(y_j) + \frac{l-i}{l}\prod_{j=1}^{i} P_f(y_j). \qquad (20)
\end{aligned}$$

□

REFERENCES

[1] N. Van Saberhagen, "Cryptonote v 2.0," 2013.

[2] A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A traceability analysis of monero's blockchain," in *European Symposium on Research in Computer Security*. Springer, 2017, pp. 153–173.

[3] E. Fujisaki and K. Suzuki, "Traceable ring signature," in *International Workshop on Public Key Cryptography*. Springer, 2007, pp. 181–200.

[4] M. Möser, K. Soska, E. Heilman, K. Lee, H. Heffan, S. Srivastava, K. Hogan, J. Hennessey, A. Miller, A. Narayanan *et al.*, "An empirical analysis of traceability in the monero blockchain," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 3, pp. 143–163, 2018.

[5] D. A. Wijaya, J. Liu, R. Steinfeld, and D. Liu, "Monero ring attack: Recreating zero mixin transaction effect," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, Aug 2018, pp. 1196–1201.

[6] Z. Yu, M. H. Au, J. Yu, R. Yang, Q. Xu, and W. F. Lau, "New empirical traceability analysis of cryptonote-style blockchains," in *Financial Cryptography and Data Security*, I. Goldberg and T. Moore, Eds. Cham: Springer International Publishing, 2019, pp. 133–149.

[7] A. Mackenzie, S. Noether, and M. C. Team, "Improving obfuscation in the cryptonote protocol," *Monero research lab report MRL-0004*, 2015.

[8] "Bitcoin utxo age distribution," https://chart-studio.plotly.com/ unchained/37.embed, accessed: 2021-03-01.

[9] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, July 1948.