MONERO RESEARCH LAB

TECHNICAL NOTE MRL-0007

Sets of spent outputs

Sarang Noether*
Monero Research Lab
October 24, 2018

Abstract

This technical note generalizes the concept of spent outputs using basic set theory. The definition captures a variety of earlier work on identifying such outputs. We quantify the effects of this analysis on the Monero blockchain and give a brief overview of mitigations.

1 Introduction

Transactions in Monero generate *outputs* (sometimes called *notes* in other literature) destined for a set of recipients by consuming one or more existing outputs under the sender's control. For each spent output in the transaction, the sender chooses a collection of arbitrary outputs from the blockchain into a *ring*. The transaction includes a proof that for each ring, any of the ring's outputs is equiprobable as the spent output. A *key image* (also called a *tag* in other literature) is included to ensure that no spent output has been spent in any previous transaction.

It is important for sender anonymity that no output in a ring is otherwise known to have been spent from external information. If an output is known to be spent, an observer can reduce the effective size of the ring as an anonymity set. If this process continues with enough outputs, the true spent output may be identified. We stress that Monero outputs cannot be linked to the wallet address of the sender, providing an additional layer of protection.

At Monero's launch, senders could choose any ring size, including a ring containing only a single output; this output is obviously the true spend. With this information, it is possible to deduce other spent outputs in small rings. Later protocol upgrades added consensus-enforced minimum ring sizes that have increased over time. These increases, as well as a transition to outputs with confidential amounts, have all but eliminated the effects of these early trivial rings. However, it is possible to generate more complex sets of rings that together reveal spent outputs, even though it may not be possible to identify which transaction spent such an output.

In this technical note, we define the idea of a spent output in a general way using basic set theory. We show that this definition captures several known methods for spent output identification. Using a tool available to all Monero users, we quantify the occurrence of many spent outputs on the Monero blockchain, showing that modern transactions are essentially unaffected by them.

2 Definition

Let $\mathcal{N}$ be the (finite) set of all outputs on a blockchain. We define a *ring* as a subset of $\mathcal{N}$. A ring containing exactly n elements is an *n-ring*. We often use lowercase letters as generic outputs.

*sarang.noether@protonmail.com

Definition 1. *Let $\{R_i\}_{i=1}^n$ be a set of rings. We say each R_i is spent if*

$$\left|\bigcup_{i=1}^{n} R_i\right| = n.$$

An output is spent if it is an element of a spent ring.

Example 1. *Let $R = \{a\}$ be a 1-ring. Then the output a (and R itself) is spent.*

Example 2. *Let $R = \{a, b\}$ and $S = \{b, c\}$ and $T = \{a, c\}$ be rings. Then each output (and ring) is spent.*

3 Specific cases

Earlier work like in [4, 2, 1, 3, 5] has suggested several classes of spent outputs. We review some of them briefly and show how they fit into our definition.

3.1 Chain reaction

The so-called *chain reaction* method uses trivial rings to iteratively identify spent outputs. This method first marks all 1-rings as spent, and removes the corresponding outputs from all other rings. It repeats this process until no 1-rings remain. The initial presentations of this method were also identified in the context of an active attack, where the adversary spends many outputs in 1-rings in an attempt to identify honest users' spent outputs.

At the end of the iteration process, each spent ring contributes a single unique spent output that was the last such identified output in the ring. This means the collection of all such spent rings matches our definition.

3.2 Ring repetition

The so-called *ring repetition* method uses multiple appearances of the same ring to identify spent outputs. This method simply identifies a collection of n separate n-rings containing the same outputs, where we can conclude that the ring is spent. This analysis was initially presented as semi-cooperative attack, where an adversary generates ring repetitions of controlled outputs to signal to other adversarial users that the ring is spent.

This method trivially matches our definition.

3.3 Subset analysis

The standard Monero toolset includes an optional blackball tool that scans the blockchain and flags certain classes of spent outputs. In addition to the chain reaction and ring repetition methods, the tool can also perform a *subset analysis*. In this method, the tool iterates over each ring. For each of the $2^n - 1$ (nonempty) subsets of an n-ring R, it counts the number of occurrences of the subset as a standalone ring elsewhere. If the sum of all such counts for subsets of R is exactly n, it flags R as spent.

This method trivially matches our definition.

3.4 Other analysis

Absent other information, our definition completely captures on-chain spent outputs. However, other sources exist in practice that may be used to flag outputs as spent, either by attackers or by users who wish to avoid selecting such outputs in new rings.

- **Chain forks**: In the event of a chain fork, a user may choose to spend the same output on multiple forks. The construction of Monero key images means that each spend of the same output will yield the same key image. An observer who sees distinct rings on multiple forks with the same key image can conclude that the spent output must appear in the intersection of all such rings, which statistically is likely to reveal the spent output. Observe that this analysis is beyond the scope of our definition.
- **Output age distribution**: A variety of heuristics exist that may give an adversary a statistical advantage in guessing the spent output in a ring. For example, spend analysis on transparent blockchains suggests that recently-generated outputs are more likely to be spent than older outputs. We note that in practice, selection of non-spent ring elements according to a distribution matching expected spend patterns easily mitigates the effectiveness of this particular heuristic. There exist other heuristics that we do not consider here. Such heuristics do not inherently provide proof that a given output is spent, and are beyond the scope of our definition.

4 Mitigations

It is possible in theory for each user to scan her copy of the blockchain, identify all spent outputs using whatever information sources are available, and ensure that she does not choose spent outputs as ring members in future transactions. However, a complete set-theoretic characterization using our definition is impractical. Even the use of an integrated blackball tool that performs only a partial analysis may take several hours for a recent snapshot of the Monero blockchain, and would need to be regularly updated for maximal privacy.

Fortunately, the risk to users of spent output identification is negligible. Early chain reaction effects among small rings dissipated quickly early in the Monero blockchain's history. As mandatory minimum ring sizes have increased, the likelihood of an accidental ring union producing a set of spent outputs is vanishingly small. While an attacker could generate collections of rings maliciously designed to produce spent outputs, a non-cooperating attacker may need to perform intensive computations to detect them; further, the generating attacker can always identify her own controlled outputs regardless of their association with other rings, making such an attack unlikely to be of additional value since it costs the attacker fees.

The number of spent outputs produced from a chain fork depends highly on the number of existing outputs spent on multiple chains, and requires a large fraction of the existing network to participate. Further, modern selection algorithms for ring members strongly favor newer outputs, meaning the effects of a fork dissipate quickly over time. In practice, the combination of these effects renders them generally impractical.

To quantify these effects, we analyzed the Monero blockchain in October 2018 using the integrated blackball tool. This tool examined several classes of spent outputs:

- outputs included in 1-rings
- outputs in repeated rings (discussed above)
- outputs identified by subset analysis (discussed above)

- outputs identified by chain reaction analysis (discussed above)

We further classify these outputs by whether they use confidential amounts. Modern transactions choose only decoys that use confidential amounts. Table 1 shows the results of this analysis.

	Legacy outputs	Confidential outputs
1-ring	12147067	0
Repeated	40	5
Subset	5916927	0
Chain reaction	749688	0
Total spent outputs	18813722	5
Total outputs on chain	21850122	7445622

Table 1: Spent output analysis for Monero blockchain as of October 2018 using integrated blackball tool

While the analysis shows that 86% of all non-confidential outputs are identified as spent, 0% of confidential outputs are. Since modern transactions only use the latter type as decoys, the effects of spent output analysis on anonymity is completely negligible.

5 Conclusion

We have presented a simple set-theoretic definition that completely characterizes spent outputs on the Monero blockchain given only set information about the ring elements themselves. This definition captures and generalizes other analysis presented elsewhere. While this definition does not address external information from sources like forked chains or temporal analysis, it offers insight into the selection of outputs toward optimal spend anonymity. While a complete analysis of all spent outputs on the Monero blockchain is computationally infeasible, we quantified several known classes of spent outputs and determined that modern transactions are unaffected by them.

References

[1] Amrit Kumar, Clément Fischer, Shruti Tople, and Prateek Saxena. A traceability analysis of Monero's blockchain. Cryptology ePrint Archive, Report 2017/338, 2017. `https://eprint.iacr.org/2017/338`.

[2] Adam Mackenzie and Surae Noether. Improving obfuscation in the CryptoNote protocol. Monero Research Lab, MRL-0004, 2015. `https://lab.getmonero.org`.

[3] M. Möser, K. Soska, E. Heilman, K. Lee, H. Heffan, S. Srivastava, K. Hogan, J. Hennessey, A. Miller, A. Narayanan, and N. Christin. An Empirical Analysis of Traceability in the Monero Blockchain. *ArXiv e-prints*, April 2017.

[4] Surae Noether, Sarang Noether, and Adam Mackenzie. A note on chain reactions in traceability in CryptoNote 2.0. Monero Research Lab, MRL-0001, 2014. `https://lab.getmonero.org`.

[5] Dimaz Ankaa Wijaya, Joseph Liu, Ron Steinfeld, and Dongxi Liu. Monero ring attack: Recreating zero mixin transaction effect. Cryptology ePrint Archive, Report 2018/348, 2018. `https://eprint.iacr.org/2018/348`.