Monero v7 Mining Pool Report

Josue Sneur
<jsneur@gmail.com>
(https://github.com/jsneur)

May 5, 2018

Abstract

This report applies to Monero mainnet version 7 (v7) and showcases degraded privacy due to publicly-available metadata: mining pools' announced blocks and transactions. Simple practices are presented which allow users and mining pools to proactively maintain privacy without disclosing less metadata. Submissions are made to the blackball database and sample code is presented for scraping known spent outputs.

Contents

1 Introduction

As of Monero version 7 (`v7`, which began at block height 1546000,) a majority of the Monero hashrate is attributable to particular mining pools. There are at least seven or eight public pools with over 1% of the total global hashrate. These pools advertise various statistics—metadata—including the blocks that they have mined. All but one pool also list the payouts that they have made to their miners. The combination of output ownership and transaction authorship allow the true member of some ring signatures to be inferred beyond a reasonable doubt in some cases, degrading one of Monero's layers of privacy.

1.1 Monero's privacy layers

Monero provides stealth addresses, ring signatures, and Ring Confidential Transactions (RingCT) as privacy layers as of `v7`. These layers ensure that Monero transactions are unlinkable, untraceable, and opaque. Stealth addresses provide unlinkability by encrypting the recipients of payments such that only the recipient of a Monero transaction can detect it as addressed to them, ring signatures provide untraceability by concealing the source of a payment such that any one of a number of ring members should be equally-plausible as the actual source of a Monero transaction, and Ring Confidential Transactions provide opaqueness by encrypting a payment's input and output amounts such that a third party cannot discern anything related to a transaction's amounts other than that it did not create new coins.

1.2 Degraded untraceability of ring signatures due to poor mining pool practices

Ring signatures should conceal the real source of a Monero transaction: there is no way to discern which ring member actually made the transaction without additional information. Unfortunately, metadata announcements due to poor mining pool practices such as routine statistics advertisements can provide enough additional information to discern which ring member is the actual source of a transaction. When mining pools announce which blocks they mine and then later announce a transaction that uses one of their blocks' outputs as a ring signature member, it is possible to deduce that is is the real souce of the transaction. By identifying an output as real in one transaction, its suitability as a decoy is degraded elsewhere, reducing the effective ring size of other users' ring signatures when included in rings after they are revealed to be known spent.

1.3 Monero version 7's countermeasures against mining pool privacy degradations

Several tools were provided by the Monero `v7` upgrade that allow users to avoid or mitigate of the above potential degradations to their privacy outlined

previously. For example, users can avoid including any known spent outputs in their own ring signatures by using what is known as the blackball database, which contains every known spent output. This report presents additional submissions to the blackball database, including explanations of what metadata identifies the real member of a ring, where to collect it, and example code to scrape and analyze the metadata necessary for independent verification of these results.

1.4 A note on key image reuse and 0-decoy transactions

The impact of reckless key image reuse across blockchains and 0-decoy (0-mixin) transactions negatively impacts the privacy provided by Monero's ring signatures, and can be used in coordination of the known mining pool outputs to create a larger impact on these ring signatures. Monero noted the issues with 0-decoy transactions in a January 2015 report by the Monero Research Lab, `MRL-0004`, and 0-decoy transactions have been prohibited on the network since March 2016.[1] They are no longer a concern with transactions going forward.

Blockchain forks have the potential to degrade the untraceability of ring signatures when key images are reused on both sides of a blockchain fork without taking advantage of any of the tools provided by Monero's 'v7' upgrade. Key images may be safely (at least privately) reused across blockchain forks if care is taken to construct identical rings on both sides of the fork (a process which is outside of the scope of this report but is described in detail here.)[2] If users send funds on both sids of the fork without any such precautions, however, then they will inadvertently produce two rings that share only one member in common, thus identifying the common member as the real source of both transactions. Such outputs are identified as "known spent."

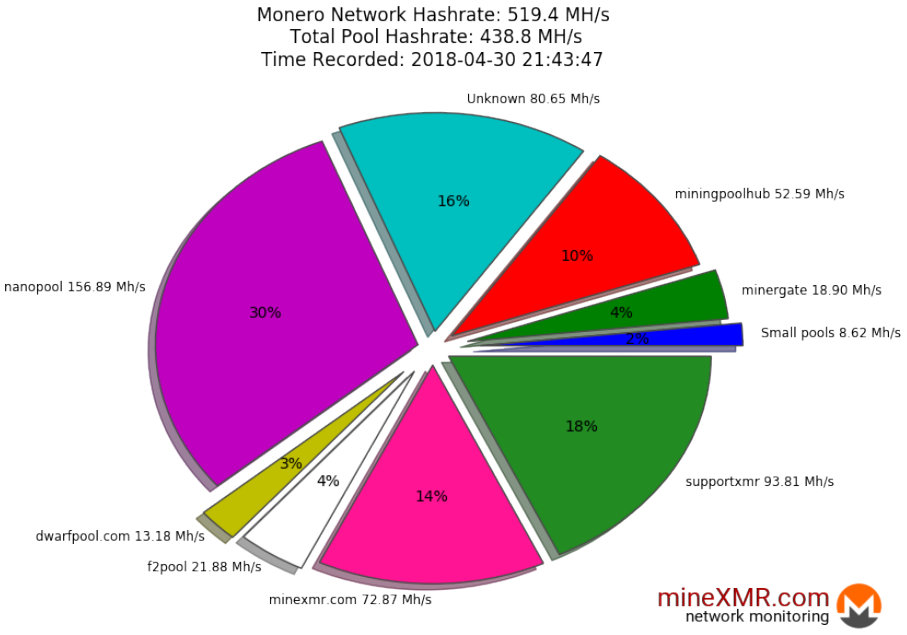
One must note that not all forks will include the opportunity to reuse ring members. Suppose a imposes a restriction to only allow 0-decoy transactions. There is no non-"reckless" way of claiming funds on such a fork; any claim degrades the privacy of Monero by rendering the associated ring signature useless and impacting transactions that use this output as a decoy. However, even if this is not the case, users can still have their privacy impacted when reusing the same ring member set. If the ring member mitigation tool is not widely used, the real output can be revealed, removing the effectiveness of the ring signature on both chains and impacting transactions that use this output as a decoy.

[1] `MRL-0004`, `https://lab.getmonero.org/pubs/MRL-0004.pdf`.

[2] "How can individuals safeguard themselves and the community against a key reusing fork?", `https://monero.stackexchange.com/questions/7826/how-can-individuals-safeguard-themselves-and-the-community-against-a-key-reusing`.

2 The State of the Hashrate

As of `v7`, a majority of the network hashrate is attributable to public mining pools.[3] The top 8 public Monero pools (in descending order of hashrate) are: Nanopool, SupportXMR, mineXMR.com, Mining Pool Hub, F2Pool, MinerGate, DwarfPool, and MoneroHash. These pools represent over 80% of the combined global hashrate. They all announce enough information to discern some outputs as known spent.[4]



Global Monero Mining Network Hashrate Distribution

2.1 Software centralization

TODO: List the most common mining pool stratum servers and GUIs and how code reuse enabled scraping of the data used to prepare this report. *Thanks, poolui!*

[3] See Figure 1 from `http://minexmr.com/pools.html`; however, independent verification of this information is a topic for future work and is possible as long as mining pools publicly disclose either a reported hashrate or at least their found blocks.

[4] All of the pools announce their finds (blocks) and all but one (Nanopool) list all of their payments (transactions.) Nanopool does not directly announce all of their payments, but still announces enough information to identify some outputs as spent, as detailed in section **??**. What portion of their total payments are attributable is not known at this time and is a topic for future work.

Table 2.1: Top 8 Public Monero Mining Pools

Pool name	API format	API endpoint
Nanopool	`naopool`	`https://api.nanopool.org/v1/xmr`
SupportXMR	`polui`	`https://supportxmr.com/api`
mineXMR.com	`noe-cryptonote-pool`	`https://p5.minexmr.com`
Mining Pool Hub		
F2Pool		
MinerGate		
DwarfPool		
MoneroHash		

3 Metadata collection

Let O be the set of a pool's outputs and T be the set of a pool's transactions; if any of their outputs o ($o \in O$) are used as a ring member in a later transaction t ($t \in T$),) then that output o is probably the real member of the ring signature: it is "known spent."

3.1 Scrape mining pool APIs for blocks (coinbase outputs)

All pool API formats make it easy to scrape mining pools for their blocks. Suffix the "API call" column (where `N`/is an integer) to the appropriate "API endpoint" column from table.2.1

Table 3.2: Mining pool API endpoints for a mining pools' found blocks

Pool format	Found blocks API
poolui	`pool/blocks?limit=N`
nanopool	`pool/blocks/N`
mineXMR.com	`stats`

3.2 Scrape mining pool APIs for transactions

All pool API formats make it easy to scrape mining pools for their transactions except for Nanopool. Nanopool's scraping is the topic of section 3.2.1. For all other API formats, just suffix the "API call" column to the appropriate their "API endpoint" column.

Table 3.3: Mining pool API endpoints for transactions

Pool format	Transaction history API
poolui	`pool/payments?limit=N`
nanopool	*none*
mineXMR.com	*to do*

3.2.1 Nanopool

4 Mitigation

TODO: Pool operators: either announce less information *or* churn prior to paying out to miners.

4.1 Blackball database submissions

TODO: Describe the https://xmreuse.daemon.network API for querying blackball database submissions

5 Future work

5.1 Maintenance of node-cryptonote-pool as node-monero-pool?

6 Conclusion

7 References