Attacking Zcash Protocol For Fun And Profit
Whitepaper Version 0.1

Duke Leto + The Hush Developers[†]

May 5, 2020

Abstract.

This paper will outline, for the first time, exactly how the "ITM Attack" (a linkability attack against shielded transactions) works against Zcash Protocol and how Hush is the first cryptocoin with a defensive mitigation against it, called "Sietch". Sietch is already running live in production and undergoing rounds of improvement from expert feedback. This is not an academic paper about pipedreams. It describes production code and networks.

We begin with a literature review of all known metadata attack methods that can be used against Zcash Protocol blockchains. This includes their estimated attack costs and threat model. This paper then describes the "ITM Attack" which is a specific instance of a new class of metadata attacks against blockchains which the author describes as "Metaverse Metadata" attacks.

The paper then explains Sietch in detail, which was a response to these new attacks. We hope this knew knowledge and theory helps cryptocoins increase their defenses against very well-funded adversaries including nation states and chain analysis companies.

A few other new privacy issues and metadata attacks against Zcash Protocol coins will also be enumerated for the first time publicly. The ideas in this paper apply to all cryptocoins which utilize transaction graphs, which is to say just about all known coins. Specifically, the Metaverse Metadata class of attacks is applicable to all Bitcoin source code forks (including Dash, Verge, Zerocoin and their forks), CryptoNote Protocol coins (Monero and friends) and MimbleWimble Protocol (Grin, Beam, etc) coins but these will not be addressed here other than a high-level description of how to apply these methods to those chains.

> In privacy zdust we trust.
>
> If dust can attack us, dust can protect us.
>
> – Sietch Mottos

Keywords: anonymity, zcash protocol, cryptographic protocols, zk-SNARKs, metadata leakage, deanonymization, electronic commerce and payment, financial privacy, zero knowledge mathematics, linkability, transaction graphs, shielded transactions, blockchain analysis .

Contents

[†] myhush.org, https://keybase.io/dukeleto, F162 19F4 C23F 9111 2E9C 734A 8DFC BF8E 5A4D 8019

1 Introduction

2 Metadata Analysis of Zcash Protocol Blockchains: Basics

2.1 Concepts and Definitions

This paper will be concerned with **transaction graphs**, which we define in the traditional mathematical sense, of a set of nodes with a set of vertices connecting nodes. In cryptocoins these always happen to be directed graphs, since there are always funds which are unspent becoming spent, i.e. a direction associated with each transaction. This direction can be mathematically defined using the timestamp of the transaction. Inputs are unspent at the time of the transaction and outputs are spent at the time of the transaction.

There is a great deal of mathematical history devoted to the study of **graph theory** that has not been applied to blockchain analysis, mostly because there was no blockchains to analyze just a few years ago and there was no financial profit in studying the data. That has obviously drastically changed.

This paper will be primarily concered with **shielded transaction graphs** which are **directed acyclic graphs (DAGs)**. A ***shielded*** transaction does not reveal the address of Alice, nor Bob, nor the amount transacted but it does leak a large amount of metadata at the protocol level, which is not rendered by block explorers nor well understood by the industry.

A ***shielded*** transaction has at least one ***shielded*** address, referred to as a ***zaddr***.

We here concern ourselves only with **Zcash Protocol** which allows us to specify a coherent language and symbols to describe the new ITM ***zaddr*** linkability attack and mitigations against it. All techniques here could technically also be used against transparent blockchains, but since they leak all the useful metadata already, it would serve no purpose. These new attacks can be thought of as "squeezing" new metadata leakage from zaddrs out of places that nobody thought to look.

For those coins which only have a transaction graph at the network p2p level but not stored on their blockchain (such as MimbleWimble coins), it does raise the bar and attack cost. Since nation-states and are not cost-sensitive and obviously have a vested interest to de-anonymize all blockchains, MW coins are not immune to these new attacks being applied. A transaction graph still exists and so the core concepts here can be applied.

2.2 Types Of Shielded Transactions

There are many types of shielded transactions, mirroring the complexity of transparent transactions in Bitcoin Protocol. Here we introduce a convention for describing transactions.

- A fully shielded transaction T with change $T : z \to z, z$
- A fully shielded transaction T with no change $T : z \to z$
- A shielded transaction T with transparent change $T : z \to z, t$
- A dehielding transaction T with change $T : z \to t, z$
- A dehielding transaction T with no change $T : z \to t$
- A shielding transaction T with no change $T : t \to z$
- A shielding transaction T with shielded change $T : t \to z, z$
- A shielding transaction T with transparent change $T : t \to z, t$

The above summarizes the most common transactions. Now say we want to describe a transaction which sends to 5 ***zaddrs*** and 3 transparent addresses with no change: $z \to z, z, z, z, z, t, t, t$. To describe very large transactions subscripts can be used : $z \to z_{52}, t_{39}$.

An individual transaction T is a sub-graph of the full transaction graph $T \subset \mathbb{T}$ with vertex count of one.

3 Metadata Analysis of Zcash Protocol Blockchains: Advanced

3.1 Active vs Passive Attacks/Analysis

3.2 Timing Analysis

3.3 Value Analysis

3.4 Fee Analysis

3.5 Input/Output Arity Analysis

3.6 Dust Attacks

3.7 Exchanges and Mining Pools

3.8 What does the explorer not show?

4 De-anonymization techniques literature review

4.1 Applications to new Shielded-only Chains

5 ITM Attack: z2z Transaction Linkability

The **ITM Attack** specifically "attacks" a transaction $T : z \rightarrow z, z$, i.e. a fully-shielded Zcash Protocol transaction which has the highest level of privacy. First we describe the definition of the attack success, if any of the following datums can be ascertained:

- The value in the ***zaddr***sending funds.
- The value any of the ***zaddrs***receiving funds.
- The value of any ShieldedInputs spent in the transaction.
- A range of possible values being sent to any ***zaddr***, such as between 0.42 and 1.7 (with error estimate)
- A range of possible values stored in the sending ***zaddr***.

If any of the above metadata can be "leaked", the attack is a success. We note that this attack is completely passive in it's core, but can be greatly improved by adding active components "to taste". This is why metadata leakage attacks such as this can be thought of a method of analysis or an outright attack.

The **ITM Attack** takes transaction id's and ***zaddrs***as input, or other OSINT which is readily available on Github, Twitter, Discord, Slack, public forms, mailing lists, IRC and many other locations. With these public resources, the **ITM Attack** can bridge the gap from theoretically interesting attack to actually de-anonymizing a ***zaddr***to it's corresponding social media accounts, email addresses, IP addresses, location data and more.

This attack is not for weekend warriors or individuals with small budgets and is not cost-effective for attacking a single ***zaddr***. It's best suited for the largest players in The Great Game, i.e NSA, GCHQ and friends. It's highly likely they already utilize analysis and attacks described in this paper.

Only the most well-funded private blockchain analysis companies will be able to afford the infrastructure for this attack, but once the data is "mined" it is a commodity that can be bought and sold to those with less resources.

The ITM is an additional "layer" of analysis that can be overload on top of all other types of analysis, and in that way it has the potential to "finish" a lot of "partial de-anonymizations", i.e. places where blockchain analysis provides some data, but not enough to fully de-anon. When added to timing analysis, amount analysis and fee analysis, it can identify that certain ***zaddrs***being involved in many transactions and their approximate input and output values. This data is not available any other way and exact values are not very important.

If a blockchain analyst can ascertain a transaction involves at least 1M USD in value versus a few pennies of value, that directly the course of analysis and investigation. Perfect de-anonymization is not needed and in practice does not matter. Software enabled with data from ITM analysis will be able to identify transaction outputs as having certain ranges of values and potentially their associated zaddrs from OSINT data.

5.1 ITM Attack: Assumptions

Fully working example code is left as an exercise to the interested blockchain analysis company. We shall describe the attack in enough detail for experts to verify our claims and for developers to implement attacks and or defenses, in the spirit of radical transparency.

We assume an attacker has at least 100,000 USD in funds to dedicate to the operation of studying one particular Zcash blockchain. Most of this cost is in the purchase of a GPU/FPGA farm to crunch data. Blockchains with more history and larger shielded pools will be more costly to study.

We note that this attack is not financially feasible as a one-off, it's a methodology to study an entire blockchain which can then be indexed and search for potentially valuabledata. Blockchain anlaysis companies and the IC are strategically positioned to use this information with the least cost, since they already have massive infrastructure to support this new dataset.

5.2 ITM Attack: Defeating *ZK-SNARKs*

We can think of this attack as a "defeat" of zero-knowledge mathematics only in practice, not in theory. Many qualifications are needed. We in no way "broke" the mathematics of ***ZK-SNARKs***, we are taking advantage of how ***ZK-SNARKs***are being used in higher level protocols, i.e. the Zcash Transaction Format Protocol and it's associated consensus rules.

So ***ZK-SNARKs***are sound and we have not actually leaked **knowledge** directly from a **zero-knowledge proof**, that is mathematically impossible. We have leaked knowledge from how these proofs are used in the larger system called Zcash Protocol, itself an extension of Bitcoin Protocol which notoriously leaks metadata.

5.3 ITM Attack: Infrastructure

This attack requires storing a lot of intermediate data in addition to the raw blockchain data and data storage costs are likely the number two expense after computing power. It is possible renting compute power can lower computing expenses but will not lower data storage costs. If one is analyzing a blockchain of $Bbytes$ then a reasonable estimate is that $100 * Bbytes$ of intermediate storage will be needed to analyze the data and then a highly compressed version of the final useful data can likely be stored in $B100bytes$ or less. That is, the final datasize will be much smaller than the input data but our intermediate will likely be two orders of magnitude larger.

Assume we have a simulated blockchain at block N, held in stasis and the analyst has their own mining hashrate to "push" the chain forward by it's own defined consensus rules. This can be accomplished by blocking all outside nodes and only connecting to the local hashrate.

We also assume the analyst can easily "spin up" a blockchain at a certain block height and try a new change to extract new data. This is trivially possible with virtual machine images, docker containers and/or Git, and is left as an exercise to the motivated blockchain analyst.

5.4 ITM Attack: Consensual Oracles

We now analyze a specific $T : z \rightarrow z, z$ at a speficic block height H which defines a specific **shielded pool** containing unspent shielded outputs and their associated metadata, such as **Merkle Tree** data.

Very specifically, the simulation will use the **SaplingMerkleTree** internal Zcash Protocol datastructure defined in src/zcash/IncrementalMerkleTree.hpp . The ITM Attack focuses on this data structure but others can and should be explored as metadata oracles, such as the **SaplingWitness** data.

At any given block height H a shielded "note" or **zUTXO** is either spent or unspent. Just like transparent **UTXOs**, a **zUTXO** can be spent from the mempool, i.e. the output of a transaction in this block can be spent by another transaction.

Different implementations of Zcash Protocol may react differently to spending zfunds from the mempool and so that is definitely a potential area of research.

Known Sapling commitments/anchors are "swapped" into the SaplingMerkleTree one at a time, in an attempt to identify if they are being spent. If the new solution tree is invalid, then the data that was added caused it to become an invalid tree for a particular reason and that particular reason is conveniently given when consensus-level errors are emitted in Bitcoin and Zcash Protocols. These errors have their own error codes and provide a wealth of information leakage to the aspiring analyst. By trying various known bits of data and analyzing the exact consensus error codes emitted, information is leaked.

6 Metaverse Metadata Attacks

The ITM Attack is a special case of what we name **Metaverse Metadata Attacks**, applied to Zcash Protocol shielded transaction graphs.

The term **Metaverse** is appropriate because alternate possible blockchain histories can be simulated to see what consensus rules would have produced. By meticulously changing one piece of data at a time, the analyst can use the consensus rules at that moment in blockchain history as an **oracle**. In this sense, **Metaverse** attacks can be classified as **consensus oracle attacks**, similar to **compression oracle** attacks and **padding oracle** attacks such as BREACH and CRIME against TLS.

7 Sietch: Theory

The ITM Attack relies on the fact that the most common shielded transaction on most currently existing Zcash Protocol blockchains have only 2 outputs $T : z \rightarrow z, z$ and the basic fact that if some metadata can be leaked about one output, if it's spend or it's range of possible values, it provides a lot of metadata on the other output as well.

If there were 3 outputs, then there would be uncertainty involved, instead of a more direct algebraic relation such as "if one output had amount=5 then the other output had an amount of $total - 5$". When 3 ***zaddr***outputs are involved, knowing the value of one ***zaddr***output does not provide as much information on the value of any other particular ***zaddr***.

This principle obviously increases, as the number of outputs increases, the leakage of the amount of any one ***zaddr***input becomes exceedingly less valuable and expensive metadata to utilize.

8 Sietch: Code In Production

Sietch uses a default rule of a minimum of 7 ***zaddr***outputs in a transaction. Because the average shielded transaction does not spend the input values exactly and there is a change output, in practice the average Hush transaction has 8 ***zaddr***outputs.

This is currently not a consensus rule and only enforced at RPC layer. There are currently various implementations of Sietch in our full node and lite wallets, which use raw transactions.

9 Advice To Zcash Protocol Coins

TLDR: You probably want Sietch or something like it.

10 Special Thanks

Special thanks to jl777, ITM and denioD for their feedback.

11 References