Jumblr

A Fully Decentralized Anonymizer

Built for the Komodo Ecosystem

Whitepaper - Draft v0.4

Introductory Note from Komodo

This whitepaper is a detailed explanation of Jumblr—an open-source and fully decentralized cryptocurrency anonymizer. Jumblr is integrated into a larger open-source project known as Komodo.

Many readers of this whitepaper may be interested to understand how Jumblr fits into the overall vision of the Komodo project. Such readers are currently finding it difficult to locate relevant and simplified information regarding Komodo, due to the project's technical complexity and comparatively young age. As a team, we are working diligently to provide simplified explanations of Komodo's open-source technology, and the Jumblr whitepaper is but one step forward.

To summarize the Komodo project in one sentence:

Komodo is to blockchain technology what Linux is to operating systems.

The Komodo project is an advancement in the causes of the decentralization and open-source movements; we focus on blockchain technology.

Jumblr provides the Komodo ecosystem with the option of privacy. Use of Jumblr is not required, and users who choose to employ Jumblr need use it only as often as desired.

We are actively developing various other technologies that Komodo-ecosystem members may utilize in conjunction with Jumblr. Watch for more information about these associated projects on our website: komodoplatform.com

- The Komodo Team, November 27, 2017

Abstract

Jumblr is a Komodo technology that enables users to anonymize their cryptocurrencies. At its most basic level, Jumblr takes non-private funds from a transparent (non-private) address, moves the funds through a series of private and non-traceable zk-SNARK addresses—which disconnects the currency trail and anonymizes the funds—and then returns the funds to a new transparent address of the user's choosing. Through a connected Komodo technology, The BarterDEX, Jumblr can provide this service not only for Komodo's native coin, KMD, but also for any cryptocurrency connected to the Komodo ecosystem.

Introduction

The Option of Privacy is Essential to the Komodo Ecosystem

One primary goal of the Komodo ecosystem is to provide our users with the highest levels of security. The option to enable oneself with privacy is an inherent part of a strong security system. Privacy empowers users with the ability to make choices without being directly controlled or observed by a third-party actor. Many of humanity's most meaningful advancements in art, technology, and other human endeavors began in situations where the creator had the security of privacy in which to explore, to discover, to make mistakes, and to learn thereby.

The roots of the Komodo ecosystem stem from the seminal work of Satoshi Nakamoto and his Bitcoin protocol[1]. One of the key challenges in this technology is that the original protocol does not make any account for privacy. Therefore, in advancing blockchain technology, we created Jumblr—a privacy feature—to empower Komodo-ecosystem members with this necessary security.

Challenges for Privacy-centric Systems and the Komodo Solution

Current pathways to obtain privacy in the blockchain industry have many problems.

One of the most popular methods to obtain privacy is the use of a centralized mixing service. In this process, users send their cryptocurrencies to service providers, who then mix all the participants' coins together, and return the coins according to the relevant contributions. With this method, the most dangerous issue, among many, is that for the duration of the mixing period users lose control over their currency. The funds, therefore, are subject to theft and human error.

[1] https://bitcoin.org/bitcoin.pdf

Other decentralized coin-mixing methods, such as the coin shuffle[2], require coordinating with other human parties. This also introduces the potential for the same issues of theft and human error, and adds yet another risk: the coordination between human parties can result in the disclosure of a user's privacy.

Some cryptocurrencies support mixing as a part of the normal transaction process in an attempt to provide constant anonymization. Varying methods for randomizing these transaction-mixing patterns exist among the many different brands of such cryptocurrencies. The problem underlying these patterns is that, regardless of the amount of mixing, the data remains in the public domain for computers to analyze later. As computer-processing power grows, transactions that were formerly private can become transparent when computer power surpasses the necessary threshold. Therefore, this method of privacy suffers from a lack of permanence.

The Komodo Solution: Jumblr

Our Jumblr technology solves the aforementioned issues through a two-layered approach, relying on connected technologies in the Komodo ecosystem—The BarterDEX and our native Komodo coin (KMD). The process is managed locally on the user's machine and requires no third parties, human coordination, or other mixing services.

A Brief Explanation of The Two Foundational Technologies

Komodo Coin (KMD)

KMD is a cryptocurrency that enables users to conduct both transparent and private transactions. In developing the Komodo ecosystem, we use KMD as the native cryptocurrency for many connecting technologies. KMD thereby continually gains usefulness as more Komodo tools are built upon it, including Jumblr.

KMD Began as a Fork of Zcash

This coin began as a fork of the popular privacy coin, Zcash. As such, KMD retains the same inherent privacy features. Notable among these features are the Zcash parameters and zk-SNARK technology. These enable users to move funds on a public blockchain without leaving data for later analysis. This is one of the most powerful forms of blockchain privacy in existence, as the provided privacy is effectively permanent. The Zcash parameters and zk-SNARK technology provide the initial foundation for users to take transparent KMD funding and make it anonymous (with the assistance of Komodo's Jumblr technology). For more information about the Zcash parameters, zk-SNARK technology, and how the Komodo project implements and

[2] https://bitcoinmagazine.com/articles/shuffling-coins-to-protect-privacy-and-fungibility-a-new-take-on-traditional-mixing-1465934826/

relies upon them, please watch our website, komodoplatform.com. We are creating more materials and FAQs to answer relevant questions.

The BarterDEX

The BarterDEX is an open-source protocol designed and pioneered by the Komodo team. It allows people to trade cryptocurrency coins without a counterparty risk. The protocol is open-source and trading is available for any coin that developers choose to connect to The BarterDEX. The service fully realizes decentralized order matching, trade clearing, and settlement. The order-matching aspect uses a low-level pubkey-to-pubkey messaging protocol, and the final settlement is executed through an atomic cross-chain protocol. Like any exchange, our decentralized alternative requires liquidity, and we provide methods and incentives therein.

The parent project, Komodo, freely provides BarterDEX technology through open-source philosophy, and maintains no permanent lock nor management over the use of The BarterDEX. Furthermore, while The BarterDEX itself is necessary as we work to further the Komodo endeavor, it is also ultimately separate from Komodo. Were we to design The BarterDEX to be permanently locked to Komodo, it would make Komodo a centralized point of control. This would be contrary to the principles of decentralization and open-source development.

Iguana Core

A Komodo technology called Iguana Core is included within The BarterDEX, and is fundamental to the overall functionality of the Komodo ecosystem. Iguana Core is complex and to fully explain it would require a separate whitepaper. To briefly summarize, the Iguana codebase is a collection of customized technologies that connect the foundations of the Komodo ecosystem together. Iguana is primarily coded in the C programming language—the language of choice of our lead developer and Komodo founder, JL777.

In part, Iguana Core is a tool kit for Komodo developers that allows them to adopt complex technologies and to advance their underlying capabilities beyond their original capacities. For example, Iguana Core provides the tool that allows a Komodo application (such as a BarterDEX standalone GUI app) to hold multiple cryptocurrencies within one wallet (and even within one “smartaddress”), and to trade them on our decentralized exchange via atomic swaps. This capability empowers Jumblr to serve any cryptocurrency connected to the Komodo ecosystem.

The BarterDEX holds the largest amount of Iguana Core code, but Jumblr and its parent program, Komodod, also contain parts of Iguana Core as well. For more information on Iguana Core, please see our Komodo GitHub repository.

The Jumblr Process

Jumblr enables users to anonymize their funds. The Jumblr process is rooted in our native Komodo Coin (KMD), and the privacy features can extend thereby to any blockchain project connected to the Komodo ecosystem.

Anonymizing Native Komodo Coin (KMD)

At its most simple level, Jumblr takes non-private KMD funds from a transparent (non-private) address, moves the funds through a series of private and non-traceable zk-SNARK addresses—which disconnects the currency trail and anonymizes the funds—and then returns the funds to a new transparent address of the user's choosing.

The entirety of the anonymization process is conducted through the user's local machine(s), except for sending the data to the network for mining. Therefore, Jumblr eliminates many dangers, including the issues of theft, human error, the disclosure of user privacy through human coordination, and the unraveling of privacy by increasing computer processing power.

User Actions

The commands that initiate Jumblr exist within Komodo's foundational program on the user's local machine, Komodod. This program is included in a typical Komodo installation, and, under normal circumstances, Komodod is natively connected to the same KMD addresses accessed by the user.

Therefore, users in the Komodo ecosystem have access to Jumblr's anonymization technology without any further effort. Developers of standalone GUI applications for the Komodo ecosystem can integrate Jumblr commands into user interfaces in any desired manner.

There are two main commands, or API calls, available:

- jumblr_deposit <KMDaddress>
- jumblr_secret <secretKMDaddress>

jumblr_deposit <KMDaddress>

This command initiates the anonymization of KMD.

Before executing the command, the user prepares the funds by placing them within the chosen <KMDaddress>. So long as Komodod has access to the private keys of the <KMDaddress>, nothing further is required. The user simply executes the command "jumblr_deposit <KMDaddress>" and Jumblr begins watching for and processing any funds in the <KMDaddress>.

Note: We call a transparent address a "T address." These are fully accessible to the user, and they are the means of conducting normal transactions. All currency entering and leaving a T address is fully visible to the network.

On the other hand, we call a privacy-enabled address a "Z address," as they utilize the Zcash parameters and zk-SNARK technology. Z addresses are internal to the Jumblr process and a user typically does not directly interact with them.

The first step Jumblr takes is to move the user's funds from a T address to a Z address.

The First Step of the Jumblr Anonymization Process

Moving the funds from a transparent address to a privacy-enabled address.

T→Z

Naturally, as the T address is fully public, an outside observer can see the funds as they leave for the respective Z address. Therefore, to fully disconnect the currency trail, Jumblr then moves the funds from the initial Z address to yet another Z address.

Jumblr creates a new Z address for each individual lot.

The Second Step of the Jumblr Anonymization Process

Moving the funds from one unique and untraceable Z address to another

Z→Z

Through the technology of the Zcash parameters, zk-SNARKs, and Jumblr, the specific whereabouts of the funds are known only to the user. The user does not need to follow the movements of T→Z and Z→Z. However, for the advanced user, there are Jumblr commands available that allow for more active interaction at these stages (see the Komodo wiki for further details). One command to mention here is z_gettotalbalance. This reveals to the user the total balance they hold within all their Z addresses.

Upon executing the command "jumblr_deposit <KMDaddress>", Jumblr begins continually observing the <KMDaddress>. Should the user send more funds into their <KMDaddress> while Jumblr is already processing the previous amount, Jumblr will simply take these new funds into account, perform any necessary actions to properly adopt them into the process, and continue its course.

Jumblr includes two subcommands that allow the user to pause Jumblr manually: jumblr_pause and jumblr_resume. The user can also halt Jumblr by shutting down Komodod (and any relevant standalone GUI applications).

Once the funds have reached their final Z address(es), they lay dormant, awaiting the user's next command.

jumblr_secret <secretKMDaddress>

The user executes this command to complete the Jumblr process. Jumblr will extract all the user's hidden currency from each Z address and place the funds in a new T address, which we call the <secretKMDaddress>. This makes the funds spendable again.

The Third and Final Step of the Jumblr Anonymization Process

Moving the funds from the final Z address to the final T address

Z→T

While this <secretKMDaddress> is completely disconnected from the original <KMDaddress>, it is only as secret as the user makes it. For example, should the user make a purchase from the <secretKMDaddress> that asks a vendor to send a physical item to the user's home mailing address, the <secretKMDaddress> effectively loses any privacy. Users should therefore exercise caution in the creation and use of their <secretKMDaddress>, according to their own needs.

Additional Security Layers

Jumblr's Process of Breaking Down Funds

The method by which Jumblr breaks down and processes the funds provides yet another layer of privacy. Jumblr begins by taking the total amount in the <KMDaddress> and, if necessary, splitting it until the largest quantities are all equal to ~7770 KMD. It then breaks down the remainder into quantities of ~100 KMD, and then the remainder thereafter into quantities of ~10 KMD. Any final remainder (which would be anything less than ~10 KMD) is ignored. (Note that Jumblr also automatically extracts its 0.3% overall fee during the Jumblr process.)

Therefore, the total amount is broken down into lot sizes of ~7770 KMD, ~100 KMD, and ~10 KMD.

Jumblr's Process of Moving the Individual Lots into a Private Address

Jumblr does not immediately move each lot into a Z address. Instead, it performs its actions in a randomized pattern to optimize anonymity, using the collective of all Jumblr users in the Komodo ecosystem to blend the transactions of the crowd together.

First, all Jumblr actions throughout the ecosystem are programmed to cluster around block numbers that are a multiple of ten (i.e. blockchain height = XXXXX0). This gathers all Jumblr requests from all users for the given time period into one large group, clustered together every ten minutes (a single block generates every minute; therefore the tenth block occurs every tenth minute).

At the moment of activity, Jumblr does one of two things: it either performs the next action in the process of anonymization, or it does nothing.

Option 1: Jumblr performs the next action

When Jumblr looks at the next action, it can perform one of three possible steps:

- T→Z
    a. If the lot has yet to be moved out of the <KMDaddress>, Jumblr can move it from the first T address to the first Z address.
- Z→Z
    a. Assuming the lot is now in the first Z address, Jumblr can move it to the final Z address.
- Z→T
    a. Assuming the jumblr_secret API call is activated, Jumblr can move the lot from the final Z address to the final T address, <secretKMDaddress>.

Option 2: Jumblr does nothing

- At each turn, instead of performing any of the above steps, Jumblr can simply abstain from any action. This happens roughly half of the time.

Through these actions, Jumblr adds a layer of obfuscation on top of the Zcash parameters and zk-SNARK technology by adding privacy to the timing and movements of each step for each user.

Additional Privacy Considerations

Although the KMD anonymization process provides a measure of privacy and may appear to be sufficient, there are still more precautions a user must take. Two main attacks are available to a would-be sleuth.

The Timing Attack

In this attack, the sleuth simply studies the time the funds disappear from the <KMDaddress> and looks for funds to appear in a T address soon thereafter. If the privacy-user persistently chooses predictable timing for initiating and completing the Jumblr commands, a determined sleuth might deduce a user's <secretKMDaddress>.

The aforementioned process of grouping and randomizing the timing of movements provides one layer of security against The Timing Attack. Users thus blend the timing of their movements together, using the power of the collective to obscure their transactions from the sleuth.

However, The Timing Attack remains an issue if the user is the only person employing Jumblr for the duration of the anonymization of their funds. In this event, effectively no anonymization takes place. The sleuth can clearly see the funds leave from the <KMDaddress> and return to

the <secretKMDaddress> later. Therefore, to be effective, Jumblr requires more than one user and gains strength with higher levels of adoption. Given the growing size of the Komodo community, we anticipate that users will easily be able to overcome The Timing Attack.

The Knapsack Attack

The Knapsack Attack is somewhat similar to The Timing Attack, but as applied to amounts. For example, if there is only one KMD address that entered ~77000 KMD into Jumblr, and ~77000 KMD later emerges elsewhere, the sleuth can easily discern the user's <secretKMDaddress>.

The aforementioned process of breaking down the total amount into three equal sized lots (~7770, ~100, ~10 KMD) for all users provides one layer of security against The Knapsack Attack. Users again can blend their transactions together, using the power of the collective to obfuscate their movements.

Jumblr has another feature, Multiple Secret Addresses, that also protects against this attack. This feature is explained in the following section.

Further Security Enhancements to Combat The Timing and Knapsack Attacks

More Defense Against The Knapsack Attack: Multiple Secret Addresses

As another layer of security, users can create multiple secret KMD addresses (<secretKMDaddress>'s) and actively use them in the Jumblr process.

When using multiple <secretKMDaddress>'s, whenever Jumblr reaches the stage of Z→T for any given lot of KMD, Jumblr will randomly choose one of the <secretKMDaddress>'s for this lot's final T address. This enables the user to split their initial funding into many different <secretKMDaddress>'s, thus providing another layer of security against The Knapsack Attack.

Jumblr manages up to 777 <secretKMDaddress>'s at one time.

Further Enhancements Against The Timing Attack

The simplest and strongest defense against The Timing Attack is in the hands of the users. Recall that a user chooses the times they execute the commands jumblr_deposit and jumblr_secret. The longer a user maintains their currency within the shielded Z address(es), the more security they have against The Timing Attack. This is because the Jumblr actions of other users during the interim obfuscate the trail. We therefore encourage users who are mindful for protection against this attack to delay the period of execution between the two commands.

We also developed Jumblr to have additional inherent protections against The Timing Attack for cases where users desire a more immediate transfer. Assuming Jumblr is activated on the user's local computer, as soon as Jumblr detects a new deposit in the <KMDaddress>, it can begin the anonymization process. However, Jumblr deliberately delays its own progress in order to provide a layer of security against The Timing Attack.

Recall that all user actions are clustered around block numbers that are multiples of ten, and half the time, Jumblr decides to do nothing. Therefore, in statistical terms, although the Jumblr background process may be constantly running in Komodod, Jumblr only activates to check for pending tasks every tenth minute, and only performs tasks every twentieth minute. Thus, each hour has roughly three different moments when Jumblr will perform one of the three available actions: T→Z, Z→Z, and Z→T. This program randomizes the amount of time it takes to complete the Jumblr process.

Assuming during a given period of activity Jumblr decides to perform the action of T→Z, it begins by working through the different sizes of lots from largest to smallest—thus beginning with a ~7770-KMD lot until they are all allocated, then to the ~100-KMD lots, and finally to the ~10-KMD lots. During any individual period of activity, Jumblr will perform the T→Z movement for no more than a single lot, and then stop.

However, when Jumblr performs either of the other two actions (Z→Z and Z→T) it will make the transfers for all lots that are in play.

Through these additional securities, therefore, Jumblr defeats The Timing Attack and The Knapsack Attack, relying on the power of the Zcash parameters and zk-SNARK technology. The more participants in Jumblr, the more privacy users gain. For those who use Jumblr on a consistent basis, the 0.3% cost of utilizing Jumblr is offset by the 5% APR inherent in the Komodo Coin (KMD). Thus, for a small fee, Jumblr users can provide both themselves and their community with privacy.

Offering Privacy to Other Cryptocurrencies

Jumblr can anonymize any cryptocurrency that is connected to the Komodo ecosystem, as The BarterDEX is natively integrated. Currently, the user is required to perform the first and final steps of trading in the Jumblr process of non-KMD cryptocurrencies. In the long term, however, Jumblr is capable of fully automating the process. We await larger adoption to complete the non-KMD automation features.

The Jumblr Process Currently: Manual non-KMD to KMD Trading on The BarterDEX

Overall, to anonymize a non-KMD cryptocurrency in the Komodo ecosystem, that currency must first be traded on The BarterDEX into KMD. Once the underlying value is held as KMD in a <KMDaddress>, Jumblr can complete its work. Upon completion, the anonymized KMD is then exchanged on The BarterDEX again for the relevant non-KMD cryptocurrency, and returned to a secret address of the user’s choosing.

At present, while The BarterDEX is in its early stages, we are focusing our energies on building features that will increase overall BarterDEX usability, as Jumblr relies on sufficient users and non-KMD liquidity to provide anonymity for non-KMD currencies.

Future Capabilities: Jumblr Automates The BarterDEX Trading Process for the User

Once the Jumblr non-KMD features are created, when anonymizing non-KMD cryptocurrencies, Jumblr will simply be a client of The BarterDEX service.

When a user activates Jumblr for a non-KMD coin, Jumblr will instruct The BarterDEX to trade the non-KMD coin into transparent KMD according to the current prices. The underlying value now being in KMD, the Jumblr protocol performs the entirety of the anonymization process previously described. With the underlying value made anonymous, Jumblr will direct The BarterDEX to exchange the value back to the user's chosen cryptocurrency. Finally, Jumblr will return the final sum to a new cryptocurrency address, provided by the user at the outset of the process.

Due to market fluctuations, depending on liquidity, it is possible that a user will experience slippage in the underlying value of their non-KMD cryptocurrency. While it would be possible to prearrange the trade on The BarterDEX (thereby eliminating any slippage), there is no available method to make such an arrangement without leaking privacy information. The party performing the second half of the trade on The BarterDEX would be a central point of failure. Therefore, the most private method for non-KMD anonymization is to simply rely on the active BarterDEX liquidity providers.

The Jumblr Fee and Asset Chain

Method of Charge

Readers will note that there is a 0.3% fee, payable in KMD, for using the Jumblr service.

Half of the Jumblr fee (0.15%) is charged as a part of the initial T→Z transaction, and the other half is charged at the final Z→T transaction. There is no charge for the internal Z→Z transaction performed by Jumblr. The revenues generated thereby accrue in the following public KMD address:

- KMD Address for Jumblr Revenues: RGhxXpXSSBTBm9EvNsXnTQczthMCxHX91t

Method of Divvying Out Revenues to Holders of the Jumblr Asset

This fee generates revenues for holders of the Jumblr Asset Chain, which is connected to the Komodo ecosystem. At varying points throughout the year, these revenues are manually and equally divvied out to the 1,000,000 total Jumblr Assets. To receive a portion of the revenues, a user simply needs to hold a Jumblr Asset token in a KMD address. The share of revenue is sent automatically to this address, according to the number of tokens held. For members of the Komodo ecosystem who hold Jumblr assets, the incentive created thereby helps to offset the overall data weight Jumblr places on the Komodo ecosystem.

A Word on Risks Inherent in Jumblr and the Komodo Ecosystem

The Komodo coin (KMD), and therefore Jumblr by association, both rely on the Zcash parameters as put forth by the Zcash team. The Zcash parameters are a “zero-knowledge” form of technology. This is a powerful form of privacy, and arguably superior to other forms as it is effectively permanent. Relying on the Zcash parameters allows us to turn our creative resources to other blockchain-technology challenges, while still empowering members of the Komodo ecosystem with the option of privacy.

To create the Zcash parameters, the original Zcash developers had to create a series of keys that, when combined, created a master key that could unlock and lock the parameters. After using the master key to create the parameters, the team destroyed every individual key. The team conducted this endeavor in a public manner. We encourage interested readers to view the “Zcash Ceremony” explanation, and to search for other viewpoints as well.

To briefly summarize the security measures, the Zcash team used several layers of protection including: multi-party computation, air-gapped compute nodes, hard-copy evidence trails, a uniquely crafted distribution of the Linux operating system, and the physical destruction of each piece of hardware that held an individual key. The resulting layers of defense would be of the highest level of difficulty for an outsider to penetrate. Furthermore, the method of creation and destruction ensured that the internal security of the project was faultless, so long as at least one member of the entire Zcash team was honest.

By our observation, the team performed this endeavor with sufficient competence and due diligence. Furthermore, given the nature of the project, the longstanding reputation of the Zcash developers, and the modus operandi of their lives’ work, we believe they were properly motivated to perform the creation and destruction in a capable and honest manner.

Nevertheless, there are privacy advocates in the cryptocurrency industry who maintain a degree of suspicion over any project that requires an element of human trust. This suspicion extends to the Zcash parameters. These observers continually scrutinize the Zcash project, searching for more and more processes by which the creation ceremony could have failed. As yet, while various theories have been put forth, no actual failure in the Zcash parameters has been discovered.

In adopting the Zcash parameters, we receive frequent questions regarding how they affect the Komodo coin. The answer is that the privacy in the Komodo ecosystem is permanent, regardless of any potential fault by the Zcash team. Furthermore, we can adopt any updates the Zcash team releases to the parameters.

In the unlikely event that someone was able to retain a complete copy of the master key, the only power the holder would have, would be the ability to create new private money in our system. This holder could then trade that for transparent, spendable money. This could negatively impact the Komodo coin, and we would be required to adapt our platform. If a fault in the Zcash parameters were to be discovered, the Komodo team has various contingency

methods at our disposal to remove the Zcash parameters and replace them with a new set of parameters.

Though in Komodo we do not see this as a realistic threat, we nevertheless include the information here in our whitepaper to provide complete transparency for any user who seeks to invest their resources in the Komodo project.

Jumblr Provides the Komodo Ecosystem with Privacy

For the Komodo ecosystem to reach its full potential, the option of enhanced privacy must be available to Komodo users. Jumblr fills this demand.

Jumblr relies on The BarterDEX, KMD, and Iguana Core to connect to the Komodo ecosystem. The foundational privacy it offers is built upon the KMD coin, the Zcash parameters, and zk-SNARK technology. Additional enhancements are built into the Jumblr process to maximize user privacy, including protections against The Timing Attack and The Knapsack Attack. Through The BarterDEX and Iguana Core, these privacy features extend to any cryptocurrency connected to the Komodo ecosystem.

As more users become a part of the Komodo ecosystem, they can work together to enhance both their own privacy and the privacy of fellow ecosystem members. As the ecosystem continues to grow, there are various levels of growth the Komodo team can offer to the Jumblr asset, including automating the non-KMD Jumblr process. We look forward to receiving your feedback on this privacy-enhancing technology.