Alice's Ring
Protocol Whitepaper V 1.0

Thomas Hussenet
thomas@cypherlab.fr

Nathan Hervier
nathan@cypherlab.fr

Maxime Dienger
maxime@cypherlab.fr

Adam Dahmoul
adam@cypherlab.fr

November 2023

Abstract

This whitepaper introduces *Alice's Ring*, a privacy protocol based on ring signatures, a cryptographic concept originally introduced by Shamir, Rivest, and Tauman in 2001 [1]. Ring signatures are mostly known in blockchains for their use in Monero[2, 3] to ensure transaction anonymity. This is done by merging a true signature with indistinguishable decoys. *Alice's Ring* utilizes this technique to generate cryptographic proof of group membership in open and decentralized networks, such as public blockchains. The protocol extends ring signatures utility into cross-chain contexts, crucial for the evolution of decentralized applications. This whitepaper explains the protocol's innovative approach, underscoring its potential to enhance privacy in blockchain.

Keywords: signature scheme, ring signature scheme, signer-ambiguous signature scheme, blockchains

Contents

1 Introduction

The advent of blockchain technology marks a significant milestone in the digital era, heralding a shift from centralized trust models to decentralized frameworks. This transformation is central to reshaping the digital interaction landscape, emphasizing the critical importance of privacy and anonymity within blockchain networks.

Ring signatures[1], have been instrumental in this privacy-centric evolution. Initially conceived for anonymous secret disclosure, ring signatures have evolved to play a crucial role in ensuring transactional anonymity in blockchain networks. Notably employed in privacy-centric cryptocurrencies like Monero[3], these signatures obscure the identity of the sender in transactions by merging a genuine signature with multiple indistinguishable decoys, forming an untraceable ring that protects transaction confidentiality. Monero employs a unique ring signature scheme known as the Linkable Ring Signature[4]. This scheme is designed to identify whether two ring signatures originate from the same private key, which is crucial for preventing double-spending.

Another cryptographic primitive used for privacy preservation are zk-SNARKs. Vitalik Buterin explains in his post [5] how zk-SNARKs could be utilized to enhance privacy in applications. However, a significant technical challenge arises when implementing zk-SNARKs with secp256k1, the elliptic curve employed by Bitcoin and Ethereum. Due to this curve's non-pairing-friendly nature [6], creating efficient zk-SNARKs, which are crucial for privacy-preserving applications, on this curve is practically unfeasible.

Protocols such as Semaphore[7] and Sismo have developed methods to use zk-SNARKs for proving group membership on Ethereum. Semaphore effectively enables private voting and anonymous signaling through Zero-Knowledge Proofs (ZKPs), ensuring group membership while preventing double signaling. Sismo utilizes zk-SNARKs for privacy-preserving identity aggregation. These SNARKs, however, do not rely on the native blockchain's curve, such as Ethereum's secp256k1, but instead on babyJubJub[8], a curve more suited for pairing.

Relying on an other curve creates user friction. This necessitates users to establish accounts on this alternate curve, involving the generation and derivation of a new private key. Such additional steps could complicate the user experience, potentially hindering wider adoption due to the increased complexity and effort required from end users.

Moreover it adds a layer of trust into these systems. This is particularly crucial in blockchains that do not support a smart-contract layer, where verifying proofs relies on trusting the prover's execution accuracy. Given the nature of zero-knowledge proofs, this trust dependency could pose challenges in ensuring the security and reliability of the protocol, especially in environments where smart contract verification is not available.

Alice's Ring emerges as a novel approach in this context, transitioning from zk-SNARKs to ring signatures for group membership proof. This transition ushers in a versatile, cross-chain method enabling private and verifiable membership proofs, independent of the underlying blockchain architecture. Diverging from zk-SNARKs-based frameworks, *Alice's Ring* forgoes the need for a trusted setup and is inherently open to the entire network, relying solely on the blockchain's curve and public data. Building upon this foundation, the protocol dynamically constructs the group tailored to the specific requirements of the proof. This adaptability allows the verifier to confirm the presence of a private key within the group and to authenticate the attributes being proven, thereby ensuring the verification of specific characteristics while maintaining the privacy and anonymity of the group members. Due to the flexibility and adaptability of ring signatures across various signature schemes, the protocol inherently supports cross-chain functionality. The only difference in a proof across different blockchains lies in the elliptic curve used for the signature and the subsequent verification process.

This white paper explains *Alice's Ring* protocol, showcasing its potential to transform the landscape of blockchain privacy. By integrating ring signatures with Elliptic Curve Cryptography, the protocol addresses the interoperability challenges across various blockchain networks and significantly enhances privacy in blockchain.

2 Concepts

The following concepts are integral to the protocol:

2.1 Witness

A witness is an entity directly interacting with and conducting on-chain activities. Each witness has a public key and private key that are used for these on-chain interactions. The witness's on-chain activities contribute to the construction of anonymity sets The witness is not actively taking part in the protocol, but its uses of the blockchain is what allow the protocol to create proof.

2.2 User

Users are entities participating in the protocol. Like witnesses, users have public and private keys. The distinction is that users actively utilize the protocol by generating proofs.

2.3 Characteristic

This refers to an on-chain trait or attribute of a user, such as ownership of native coins, tokens, NFTs, or specific transaction types. Characteristics define the criteria for membership within a group on the blockchain network.

2.4 Anonymity Set

The anonymity set consists of public keys from various witness and the user's public key, forming the ring. This collection is key to ensuring the anonymity of the user, making it infeasible to identify the user's identity within the set.

2.5 Assessment

The assessment is the specific characteristic a user seeks to prove without revealing their identity. It functions as the message in the ring signature scheme and is a declaration of possessing a certain characteristic.

2.6 Proof of Membership

A proof of membership is established through a valid ring signature, comprising the signature itself, the anonymity set, and the assessment. This proof serves to verify that the signer belongs to a group with the specified characteristic, while safeguarding the signer's identity. It's worth noting that the level of privacy in the proof decreases as the assessment becomes rarer (e.g., holding 500,000 BTC), since the anonymity set diminishes in size.

2.7 Badge

A badge is a soul-bound token[9] that encapsulates the proof of membership within its metadata. It serves as a non-transferable, verifiable credential that the user possesses the proven characteristic or membership.

2.8 Verifier

A verifier verifies the validity of a proof of membership. This verification can be performed by a smart contract on the blockchain or an off-chain program. The verifier ensures the proof's authenticity by confirming the authenticity of the ring signature. Additionally, it verifies that all addresses within the anonymity set continue to meet the assessment criteria, indicating that they still possess the specified characteristic.

2.9 Summary

In summary, the protocol operates within a framework where witness interact with the blockchain network, possessing unique characteristics defined by their on-chain activities, such as holding specific tokens or engaging in particular transactions. These characteristics form the basis for creating anonymity sets, where a user's public key is pooled with others to maintain anonymity in the ring signature scheme.

When a user wishes to prove a particular characteristic without revealing their identity, they generate a proof of membership. This proof, validated by the verifier, ensures the authenticity of the claim and checks that all members of the anonymity set still hold the claimed characteristic.

Badges, as soul-bound tokens, play a crucial role in representing these proofs in a tangible and verifiable form, encapsulating the proof of membership in their metadata. Thus, Alice's Ring establishes a secure, transparent, and anonymous system for proving and verifying membership or characteristics within a blockchain network.

3 Protocol Mechanics

3.1 Address and Public Key Collection

Necessity of an Indexer for Enhanced User Experience: The protocol employs an indexer to aggregate blockchain data, a crucial step for ensuring a fast and seamless user experience (UX). The indexer's role is to efficiently gather and organize relevant data from various blockchain networks. This data aggregation is essential not only for the protocol's functionality, but also for providing users with a responsive and intuitive interface. By swiftly processing and presenting necessary information, the indexer plays a pivotal role in enhancing the overall UX of the protocol.

Aggregation of Publicly Available Data: In line with the commitment to privacy and security, the protocol focuses exclusively on aggregating publicly available data. This includes information like transaction histories, wallet balances, and other on-chain activities that are openly accessible on the blockchain. By limiting data collection to public sources, the protocol ensures that it adheres to privacy standards and respects the anonymity of blockchain users.

Retrieval of Public Keys Based on Network Specificities: An integral part of data aggregation involves retrieving the public keys associated with blockchain addresses. The methodology for this retrieval varies depending on the blockchain network. For instance, on the Ethereum Virtual Machine (EVM), an 'ecrecover' function is used to extract the public key from a transaction emitted by the address in question. In contrast, on the XRP Ledger (XRPL), the public key is directly available in the transaction payload. This network-specific approach to public key retrieval ensures that the protocol can operate effectively across different blockchain architectures.

Verification Utility in Absence of Smart Contracts or Oracles: The aggregated data and retrieved public keys have significant utility in verifying user characteristics, especially in blockchain networks where certain functionalities, such as smart contracts or oracles, are absent or limited. In such scenarios, the indexer's aggregated data becomes a critical resource for validating user attributes or memberships within the protocol. This capability highlights the versatility of the protocol in adapting to various blockchain environments, ensuring that it can reliably verify user characteristics regardless of the underlying network infrastructure.

In conclusion, the process of data aggregation through the indexer, focusing on publicly available information and network-specific methodologies for public key retrieval, is fundamental to the protocol. This approach not only maintains the privacy and security of users but also ensures the protocol's adaptability and functionality across diverse blockchain networks.

3.2 Ring Signature Process

The protocol leverages the SAG signature scheme, an adaptation of the Linkable Spontaneous Anonymous Group algorithm, modified to enhance non-linkability and ensure the anonymity of group members. This signature scheme is foundational to the protocol's ability to provide private group of membership proof. The SAG algorithm employed in the protocol is describe in "Zero to Monero: Second Edition"[2].

Overview of SAG Signature Scheme
The SAG signature scheme used in the protocol is based on Elliptic Curve Cryptography. This choice aligns with the cryptographic foundations of most blockchain networks, ensuring compatibility and security in the protocol's operations.

Signature Generation Process
Let l be the order of G, the generator point of an elliptic curve.
Let m be the message to sign.
Let H be a hashing function.
Let n be the size of the ring.
Let $R = \{K_1, K_2, \ldots, K_n\}$ be a set of distinct public keys, the ring, and k_π the signer's private key corresponding to his public key $K_\pi \in R$, where π is a secret index.

The process of generating a signature in the SAG scheme involves several mathematical steps:

- **Nonce Generation:** Generate a random integer α within the range $[1, l-1]$.
- **Signer index Generation:** Generate a random integer π within the range $[1, n]$, this will serve as the signer index.
- **Random Responses Generation:** For a group with n members, generate random responses $r = \{r_0, r_1, \ldots, r_n\}$, excluding r_π, where each r_i is a random integer in the range $[1, l-1]$.
- **Challenges and Responses Calculation:** Compute $c_{\pi+1} = H(R, m, [\alpha G])$ and for each member i, calculate $c_{i+1} = H(R, m, [r_i G - c_i K_i])$.
- **Final Response Calculation:** Define the signer's response to verify α as $r_\pi - c_\pi k \pmod{l}$. The signature includes the ring of public keys R, the seed c_0, and the responses r.

Signature Verification
Known data:

- The ring of public keys R
- The seed c_0
- The responses $r = r_0, r_1, ..., r_n$
- The message m

The verifier computes the following:

1. For $i = 1$ to n, with i wrapping around to 0 after n:
   (a) $c'_i = H(R, m, [r_{i-1}G - c'_{i-1}K_{i-1}])$ if $i \neq 0$ else $c'_i = H(R, m, [r_0G - c_0K_0])$
2. If $c'_i = c_0$ then the signature is valid, else it is invalid.

3.3 Badge Issuance

Now that the indexer has aggregated the necessary public keys and the ring signature generation and verification processes have been established, let's delve into how badges are issued upon generating a successful proof.

As a reminder, the ring signature allows a user to prove their membership in a certain set of addresses with specific characteristics without revealing their exact identity. This proof is generated using the SAG scheme over a ring of public keys that match the membership criteria.

When a valid ring signature is created and verified by the protocol, it constitutes proof of the user's authenticated membership in the ring. To represent their proven attributes, the protocol then mints a unique badge.

Specifically, the protocol undertakes the following steps for badge issuance:

1. A SBT[9] is created on the blockchain to represent the validated signature proof. This SBT is known as the badge.
2. The badge is then issued and assigned to a separate address that the user has access to, rather than the address from which the ring signature was generated. This approach ensures that the proof of attributes is not tied to the same address used for generating the ring signature, thus preserving anonymity.
3. Metadata linking the badge to its signature proof is stored on IPFS. This immutable record binding proves the authenticity and accuracy of the badge's claims.

By issuing a badge linked to the signature and assigning it to a separate address, the protocol can credibly represent a user's verified membership or characteristics within a privacy-preserving framework. This, in turn, enables the development of novel applications relying on provable yet anonymous on-chain attributes.

4 Case Study

This case study demonstrates the practical application of the protocol in a scenario where a user, Alice, seeks to prove her financial credibility without compromising her privacy. By leveraging the innovative approach of ring signatures, the protocol allows Alice to assert ownership of a certain amount of cryptocurrency, in this case, Ethereum (ETH), while maintaining the confidentiality of her personal wallet address.

4.1 Proof of Asset Ownership

Alice begins by initiating a request function through a Web or Mobile App, which is part of the front-end layer of the protocol. The application layer interfaces with an indexing protocol situated in the network layer to gather a list of public keys. These public keys are associated with addresses that each hold at least 5 ETH. This information is readily available and is retrieved from the blockchain, ensuring that the process preserves the native privacy of the network.

The collected public keys form an anonymity set, a ring wherein Alice's actual public key is included, but remains indiscernible among the others. Utilizing her private key, which she accesses securely within her wallet, Alice proceeds to generate a ring signature using her private key, which is securely accessed within her wallet environment. This ring signature serves as cryptographic proof that Alice possesses at least one private key corresponding to a public key in the ring. Since all the public keys in the ring are associated with addresses that each hold a minimum of 5 ETH, the signature effectively proves that Alice holds at least 5 ETH. Crucially, the ring signature conceals which of the public keys is linked to Alice's private key, thereby maintaining her anonymity.

4.2 Cross-Chain Proof Verification via Oracles

The case study further explores the scenario where Alice wants to demonstrate the ownership of Bitcoin (BTC) on the Ethereum blockchain. To accomplish this, the protocol can incorporate the use of decentralized oracles. The oracles serve as reliable intermediaries, fetching and validating data across blockchains.

Once Alice provides a ring signature including her Bitcoin address, the oracle verifies the Bitcoin holdings and communicates this information to an on-chain verifier in the form of a smart contract. The smart contract, designed to interact seamlessly with the oracle, processes the ring signature and confirms the validity of the claim. This process not only validates the existence of the required BTC amount within the ring but also does so without revealing Alice's specific Bitcoin address.

The oracle plays a pivotal role in this cross-chain context, ensuring that the verification of asset holdings can occur even when direct on-chain validation is not possible. This feature showcases the protocol's flexibility and its capacity to function across different blockchain architectures.

4.3 Conclusion of the Case Study

In conclusion, the case study of Alice demonstrates the capability of *Alice's Ring* to facilitate private and secure verification of asset ownership across different blockchain platforms. The protocol employs a combination of real-time data indexing, ring signature cryptography, and decentralized oracles to deliver a versatile and privacy-preserving solution. It exemplifies the potential of ring signatures to maintain privacy in financial interactions and validates the practicality of the protocol for real-world applications.

5 Limitations and Future Work

5.1 Non-Linkability

The first version of *Alice's Ring* embraces a non-linkable signature scheme, providing an added layer of privacy for attestations. While this feature is advantageous for certain proof scenarios, such as attestations, it introduces a potential vulnerability in more complex applications like bridges or voting systems. The absence of linkability [4, 2] could pave the way for double-spending concerns in these intricate projects. Recognizing this limitation, future versions of the protocol will incorporate linkability, offering a more versatile foundation for a broader range of applications while maintaining the necessary privacy measures.

5.2 Integrity and Temporality of the Proof

Presently, the validity of a proof in *Alice's Ring* is determined at the time of its creation. However, a crucial consideration is the dynamic nature of blockchain addresses. If an address within the ring loses the specified characteristic after the proof is generated, the proof becomes invalid. In a forthcoming version, the protocol will introduce a mechanism for periodic checks, perhaps on an hourly or daily basis. This enhancement ensures the ongoing validity of proofs by confirming that all addresses in the ring maintain the required characteristics at regular intervals.

5.3 Proof Complexity

The complexity and rarity of the characteristic being proved introduce an interesting challenge in the depth of the proof. When the characteristic is rare, resulting in fewer addresses having the desired trait, the anonymity set becomes smaller. Consequently, the depth of the proof, represented by the size of the anonymity set, is reduced. This relationship between the rarity of the characteristic, the size of the anonymity set, and the depth of the proof will be a focal point in future developments of the protocol. Strategies to address this challenge and maintain the protocol's effectiveness across a spectrum of characteristics will be explored.

5.4 Trust Requirements

One key aspect of *Alice's Ring* is its commitment to privacy. In terms of trust, the protocol relies solely on signatures and public data, aligning with the trust model inherent in the blockchain layer. As the protocol operates within the established trust parameters of blockchains, there is no additional trust requirement beyond what is expected at the blockchain layer. This emphasis on utilizing only verifiable and publicly available information ensures that the trust model of *Alice's Ring* remains in harmony with the decentralized ethos of blockchain networks.

6 Acknowledgement

We are grateful for the subsequent grant received in June 2023 to continue advancing the work. Additionally, we extend our gratitude to Henri Lieutaud and Cyril Grunspan for providing support, advice and reviews throughout our research process. We also thank the researchers and instructors at the Pole Leonard de Vinci who contributed feedback and encouragement as the project has progressed. A special thanks as well to the KRYPTOSPHERE organization for allowing us to develop our ideas within a community so generous with guidance.

References

[1] Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. *Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139; Computer Science department, The Weizmann Institute, Rehovot 76100, Israel*, 2001.

[2] Koe, Kurt M. Alonso, and Sarang Noether. *Zero to Monero: Second Edition*. Monero, 2 edition, 2020.

[3] Monero. `https://www.getmonero.org/`.

[4] Joseph K. Liu, Victor K. Wei, and Duncan S. Wong. Linkable spontaneous anonymous group signature for ad hoc groups. *Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong; Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong*, 2004.

[5] Some ways to use zk-snarks for privacy. `https://vitalik.eth.limo/general/2022/06/15/using_snarks.html`.

[6] Stéphane Vincent. Exploring pairing based cryptography. *Sikoba Research*, 2018.

[7] Semaphore whitepaper. `https://semaphore.pse.dev/whitepaper-v1.pdf`.

[8] Baby-jubjub elliptic curve. `https://github.com/iden3/iden3-docs/blob/master/source/iden3_repos/research/publications/zkproof-standards-workshop-2/baby-jubjub/Baby-Jubjub.pdf`.

[9] Erc-4671: Non-tradable tokens standard. `https://eips.ethereum.org/EIPS/eip-4671`.