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

The indexer is entrusted by the Alice's Ring protocol to perform critical data aggregation tasks. It operates as a trusted intermediary, systematically collecting data necessary for the protocol's functionality. This encompasses the aggregation of public keys, the monitoring of wallet balances, and the tracking of transaction histories. By performing these tasks, the indexer ensures that the protocol maintains an up-to-date view of the blockchain state, which is vital for executing ring signatures and managing badge issuances effectively.

Formal Specification of Data Collection Methods

1. **Public Key Retrieval:** The indexer is tasked with retrieving public keys associated with specific blockchain addresses. This process is tailored to the intricacies of various blockchain networks. For instance:
   - On networks following the Ethereum Virtual Machine (EVM) model, the indexer utilizes the 'ecrecover' function to extract public keys from transactions initiated by the addresses.
   - On the XRP Ledger (XRPL), the indexer directly accesses the public key within the transaction payload.
2. **Balance and Transaction History Monitoring:** The indexer continuously monitors the blockchain to update the balance and transaction history of each address. This monitoring is crucial for assessing the eligibility of participants in the ring signature process and for the accurate issuance of badges. The method of monitoring is designed to be efficient and timely, ensuring that the protocol can respond dynamically to changes on the blockchain.

Enhancing Protocol Security and User Experience
The indexer's efficient data aggregation and organization significantly contribute to the Alice's Ring protocol's security and operational efficiency. By providing a reliable and up-to-date data foundation, the indexer supports the accurate execution of ring signatures and the issuance of badges. Furthermore, its role in enhancing the user experience cannot be overstated. Through its rapid processing and presentation of blockchain data, the indexer facilitates a seamless and responsive interface for users, underpinning the overall usability and accessibility of the Alice's Ring protocol.

Clustering Logic for Ring Formation
An feature of the Alice's Ring protocol's indexer is its clustering logic, designed to enhance the security and anonymity of ring signatures. This logic employs

algorithms to group addresses based on similarities in their transaction patterns, asset holdings, and balance movements. The criteria used for clustering include:

- **List of Assets:** Addresses are grouped based on the types of assets they hold, ensuring that members of a ring share similar asset portfolios.
- **Balance of Each Asset:** The indexer considers the balance of each asset within addresses, clustering those with comparable asset quantities to maintain plausible deniability within the ring.
- **On-Chain Activity:** The frequency and nature of on-chain activities are analyzed to cluster addresses with similar transaction behaviors, further obfuscating the origins of a specific transaction.

This clustering mechanism ensures that when an address is utilized in a ring signature, it bears a resemblance to other addresses in the ring, making it challenging for observers to deduce the true initiator of a transaction. To prevent potential attacks where an adversary might attempt to infer the user's public key by monitoring API requests and responses, the indexer incorporates a temporal delay before reusing an address in a different ring. This safeguard ensures that even if an attacker could observe the ring composition, the interval between reuse adds a layer of complexity that significantly mitigates the risk of deducing individual addresses' identities.

Indexer API: High-Level Overview
When a user wishes to generate such a proof, the front-end interfaces with the API, submitting a request that includes the following arguments:

1. **Characteristic to Prove:** This specifies the particular attribute or characteristic the user intends to prove membership of, without revealing their specific identity within the group.
2. **Number of Addresses Needed in the Ring:** This dictates the total number of addresses that should be included in the ring to maintain the desired level of anonymity and security.

Upon receiving these parameters, the API engages the indexer's advanced data processing and clustering logic to construct an appropriate ring. The ring, comprising a set of blockchain addresses that collectively satisfy the specified characteristic and count, is then returned to the front-end. This ring serves as the foundation for generating a ring signature that conclusively proves the user's group membership while preserving their anonymity.

3.2 Ring Signature Process

The protocol leverages the SAG signature scheme, an adaptation of the Linkable Spontaneous Anonymous Group algorithm, modified to support 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 digest 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 $r = \{r_1, \ldots, r_n\}$, 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 challenge c_1, and the responses r.

Signature Verification
Known data:

- The ring of public keys R
- The seed c_1
- The responses $r = r_1, r_2, ..., r_n$

- The message m

The verifier computes the following: For $i = 1, 2, \ldots, n$ iteratively compute, replacing $n + 1 \rightarrow 1$,

$$c'_{0,i+1} = H_n(R, m, [r_i G + c_i K_i])$$

If $c'_1 = c_1$ then the signature is valid. Note that c_1 is the last term calculated.

3.3 Badge Issuance

With the establishment of the indexer's role in aggregating necessary public keys and the framework for generating and verifying ring signatures, we now explore the process of badge issuance following a successful proof of membership.

A ring signature enables a user to demonstrate their affiliation with a group of addresses that share specific attributes, without disclosing their individual identity. This proof utilizes the Spontaneous Anonymous Group (SAG) signature across a selection of public keys that align with the requisite membership traits.

Upon the generation and validation of a ring signature by the protocol, the user's authenticated membership within the group is confirmed. To symbolically represent this membership and the attributes verified, the protocol mints a unique digital badge. The process of badge issuance unfolds through the following steps:

1. A Soulbound Token (SBT), as referenced in EIP-4671, is minted on the blockchain. This token, referred to as a badge, embodies the validated signature proof, serving as a digital representation of the user's proven attributes or membership. *Soulbound Tokens* are non-transferable blockchain tokens that represent a user's credentials, achievements, or affiliations. Unlike typical tokens, SBTs are "bound" to a user's digital identity, making them a perfect tool for representing non-transferable proofs of membership or attributes.

2. To maintain anonymity, the badge is assigned not to the originating address but to a distinct address under the user's control. This separation ensures that the attribute proof remains disconnected from the address utilized in the ring signature process.

3. Metadata that establishes a connection between the badge and its underlying signature proof is securely stored on the InterPlanetary File System (IPFS). This ensures an immutable linkage that certifies the badge's claims to authenticity and validity. *The InterPlanetary File System* is a decentralized storage solution that enables the permanent and immutable storing of data across a distributed network of computers. This technology is used to securely and reliably store the metadata associated with badges, ensuring that the proof of a user's attributes is both accessible and resistant to tampering.

This approach to badge issuance, which meticulously links a badge to its corresponding signature while assigning it to an independent address, enables the protocol to authentically and privately signify a user's verified membership or characteristics. The implications of this mechanism are profound, paving the way for new applications that depend on verifiable, yet anonymous, on-chain attributes.

Understanding Badge Metadata Each badge issued within the Alice's Ring protocol contains a set of metadata that provides detailed information about the proof of membership or attribute it represents. The metadata fields encapsulated within a badge are as follows:

- **currency:** Specifies the type of currency associated with the badge, such as "BTC". This field indicates the blockchain asset relevant to the badge's proof.

- **minimumAmount:** Represents the minimum amount of the specified currency that is considered for proving membership or an attribute. For example, "1000000" signifies the minimum balance threshold in the context of the badge. This value is always expressed in the smallest unit of the currency (e.g., satoshis for BTC or wei for ETH).

- **network:** Indicates the blockchain network on which the currency operates. This detail specifies the environment of the badge's associated currency, such as Bitcoin, Ethereum, or others, providing crucial context for interpreting the badge's currency and minimum amount fields.

- **proverAddress:** Contains the blockchain address of the user who is proving their membership or attribute. This address is integral to the verification process of the badge.

- **customMessage:** Allows for an optional message to be included within the badge, providing additional context or information related to the proof. This field can be used for arbitrary data that enhances the understanding or utility of the badge.

- **createdAtBlock:** Indicates the blockchain block number at which the badge was created. This provides a timestamp of the badge's issuance in terms of blockchain history, offering traceability and immutability.

- **signature:** Contains the ring signature itself, which is a cryptographic proof that verifies the prover's claim without revealing their identity. The signature ensures the authenticity and non-repudiation of the badge.

- **ringSize:** Specifies the number of addresses included in the ring used for generating the signature. This field indicates the level of privacy and security afforded by the badge, as a larger ring size generally enhances anonymity.

This metadata is crucial for verifying the authenticity and accuracy of the badge's claims. By including such detailed information, the Alice's Ring protocol ensures that each badge serves as a reliable and transparent representation of the user's proven attributes or membership within a privacy-preserving framework.

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`.