Introduction to Ring Signatures

Marc Cosgaya Capel

Master's Degree in Cybersecurity, UPC

2022

Table of Contents

1. Introduction.....3
2. Combining Function.....3
3. Signing.....3
4. Verifying.....4
5. Security Analysis.....4
6. Improvements on Efficiency.....5
7. Applications.....5
    7.1. Traceable Ring Signatures.....5
    7.2. Linkable Ring Signatures.....6
    7.3. Threshold Ring Signatures.....6
8. Conclusion.....6

1. Introduction

Imagine that a high-ranking White House official wants to leak some information about the President. On the one hand, the official could send the information anonymously, but no reasonable person would believe that it comes from the staff. On the other hand, The official could send a signed message, but then anyone would know that it was them who leaked the information. So, is there a way to prove that a message was signed by some insider without telling exactly who?

A similar problem was presented in the Ring Signatures paper by Rivest et al. in 2001.[1] The paper demonstrated a way of leaking information while preserving anonymity. The basis of this technique is the use of a set of public keys, that obfuscate the signer's identity, and the private/public key pair belonging to the signer. The process does not need to be interactive, as the owners of the public keys are not required to consent to the use of their key. Ring signatures are similar to group signatures, but don't rely on a group manager.

2. Combining Function

Ring signatures take advantage of the properties of a combining function *C*. This function assumes a symmetric-key, one-to-one encryption function *E*. If it is deterministic and uses permutations, this property is ensured. Also assume *D* as the corresponding decryption function for *E*. *C* uses symmetric key *k*, an initial value *v* and a list of *b*-bit strings. *C* can be defined as follows:

$$\boxed{C_{k,v}(y_1,\ldots,y_r) = E_k(y_r \oplus E_k(y_{r-1} \oplus E_k(\ldots \oplus E_k(\ y_2 \oplus E_k(y_1 \oplus v))\ \ldots))) = z}$$

The interesting property is the ability to find a specific y_s value in the middle of the list by setting a value for *z*. This is called a "ring equation". An example for finding y_2 is as follows:

$$\begin{aligned}
z &= E_k(y_4 \oplus E_k(y_3 \oplus E_k(y_2 \oplus E_k(y_1 \oplus v)))) \\
D_k(z) &= y_4 \oplus E_k(y_3 \oplus E_k(y_2 \oplus E_k(y_1 \oplus v))) \\
D_k(z) \oplus y_4 &= E_k(y_3 \oplus E_k(y_2 \oplus E_k(y_1 \oplus v))) \\
D_k(D_k(z) \oplus y_4) &= y_3 \oplus E_k(y_2 \oplus E_k(y_1 \oplus v)) \\
D_k(D_k(z) \oplus y_4) \oplus y_3 &= E_k(y_2 \oplus E_k(y_1 \oplus v)) \\
D_k(D_k(D_k(z) \oplus y_4) \oplus y_3) &= y_2 \oplus E_k(y_1 \oplus v) \\
D_k(D_k(D_k(z) \oplus y_4) \oplus y_3) \oplus E_k(y_1 \oplus v) &= y_2
\end{aligned}$$

Basically, chain decryptions and ⊕ from y_r to y_{s+1}, and then chain encryptions and ⊕ from y_{s-1} to y_1. The equation only allows one unknown variable.

3. Signing

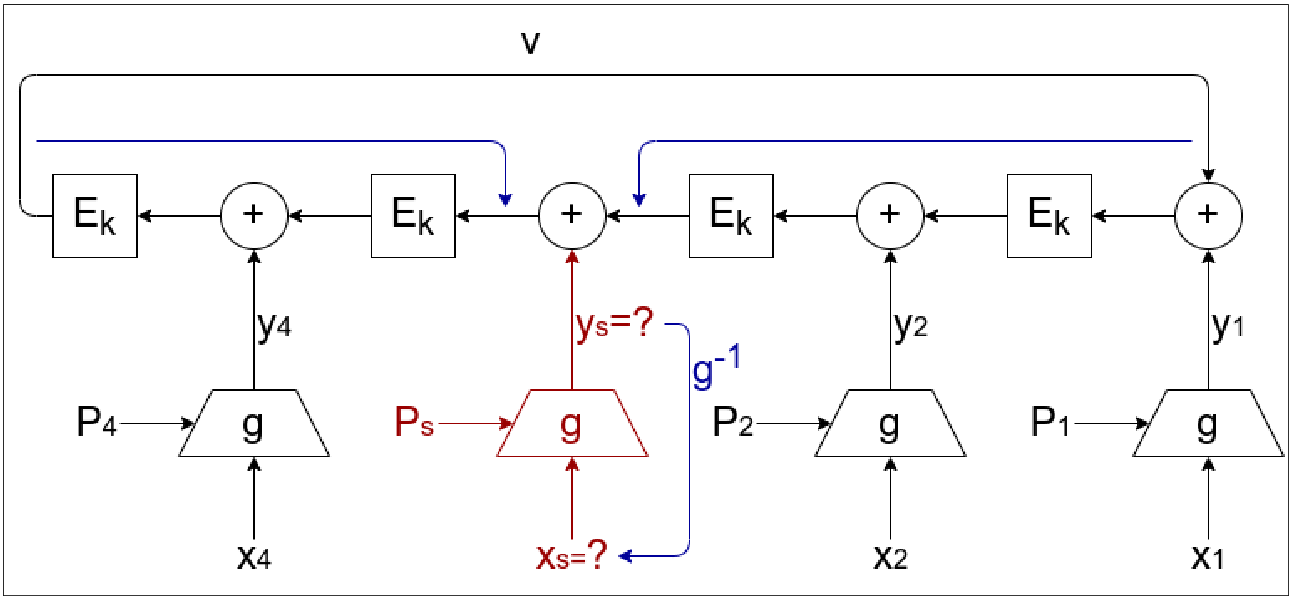
In order to sign, some steps have to be performed. Before starting, decide the size of the ring, determined by *r*. First, each y_i is the result of applying $g(x_i)$ where x_i is a random value and *g* is a trapdoor function using the public key P_i. The function can be, for example, *RSA*. Second, for *SKC* use $k = h(m)$ where *m* is the message to sign. Third, force $z = v \leftarrow \{0,1\}^b$[*] which makes a "ring"

[1] "How to Leak a Secret". Ronald L. Rivest, Adi Shamir, and Yael Tauman. November 20, 2001. <https://link.springer.com/chapter/10.1007/3-540-45682-1_32>

[*] With *RSA*, *b* is greater than the largest modulus of all the P_i. $2^b > max(n_i)$.

structure with the equation. Finally, solve it by finding the missing y_s which fulfills the equation. y_s is then decrypted using a private key, to find its corresponding x_s. This can be represented as $g^{-1}(y_s)$. Finally, output the signature as $(P_1,\ldots,P_r,v,x_1,\ldots,x_r)$.

The signing process for $s = 3$ and $r = 4$ can be imagined as follows:



A y_s needs to be found so that it fulfills the ⊕ operation at the s position. This is done by ⊕ing the previous steps with the inversion of the following steps, starting from v. Then inverting the trapdoor function with a private key to get the missing x_s that completes the ring structure.

4. Verifying

In order to verify, checking that the solved equation holds is sufficient. This proves that a private/public key pair was used to close the ring. It is achieved by calculating all $y_i = g(x_i)$ using P_i and checking that $z \stackrel{?}{=} v$ with $k = h(m)$.

5. Security Analysis

Someone must own the private/public key pair. This allows that person be the only one capable of “closing” the ring, by finding x_s, with the provided public keys and random x_i values. Note that the value of s is random and thus y_s can be at any position. Hence, it’s impossible to know which public key corresponds to the private one used to get x_s from y_s. Also, note that x_i is outputted instead of y_i. This is because, when using x_i, it proves that the P_i are valid public keys that generate the proper y_i that fulfill the equation. Moreover, it also proves that some private key is used to close the ring. In other words, **it proves that someone in the P_i list used their private key**.

The security of this scheme comes from the use of the trapdoor function g. An adversary would need to get the private key in order to close the ring. This is negligible due to the nature of the trapdoor. The scheme is EF-CMA secure because all messages use a different $k = h(m)$ in E_k. Since h is unpredictable and E_k acts like a scrambler, an adversary cannot predict the values of y_i, even if they have all other possible signature-message pairs.

However, ring signatures break the concept of non-repudiation by adding multiple possible origins to the message. Note that this is to some extent, because now the origin can be interpreted as an aggregate. This is partially solved by some variations of this scheme.

6. Improvements on Efficiency

The proposed algorithm by Rivest et al. is of asymptotic notation $O(r)$. This makes the scheme highly inefficient. Note that verification is also of $O(r)$ because the ring has to be calculated again. Since 2001, more efficient schemes have been introduced in papers.

Abe et al. in 2002 developed a ring signature based on discrete-log trapdoor functions, with signatures of $O(r/2)$ with large enough r.[2] Chandran et al. in 2007 devised a way of generating ring signatures of $O(\sqrt{r})$.[3] Maxwell et al. in 2015 defined the use multiple rings in order to save space, in a structure resembling Borromean rings.[4,5] However, Dodis et al. in 2004 claim to have found a ring signature scheme of $O(1)$.[6]

7. Applications

Going back to the example given in the introduction, the official could use the public keys of several staff members and close the ring using their private key. This allows others to see that the information came from some insider. However, no one can know for sure who did it.

A ring signature can be used as a blackmail-resistant signature.[7] Imagine a situation where Alice sent a signed message (m,σ) to Bob with some incriminating information. Now Bob wants to blackmail Alice. With a traditional signature scheme, exposing Alice would be as trivial as publishing (m,σ). However, with a ring signature Alice can shield herself against Bob by including his public key to the ring. Now Bob cannot prove to a third party that Alice signed m.

Several variants of ring signatures have been developed since 2001. In the following sections, some of these, together with their applications, will be explained.

7.1. Traceable Ring Signatures

Traceable ring signatures allow for signing messages using a tag $L = (issue, \{P_1,\dots,P_r\})$.[8] *issue* is an identifier of a topic. If the signer signs equal messages with the same tag, then it can be known that

[2] "1-out-of-n Signatures from a Variety of Keys". Masayuki Abe, Miyako Ohkubo and Koutarou Suzuki. 2002. <https://link.springer.com/chapter/10.1007/3-540-36178-2_26>

[3] "Ring Signatures of Sub-linear Size Without Random Oracles". Nishanth Chandran, Jens Groth and Amit Sahai. 2007. <http://web.cs.ucla.edu/~sahai/work/web/2007%20Publications/ICALP_Chandran2007.pdf>

[4] "Borromean Ring Signatures". Gregory Maxwell and Andrew Poelstra. 2015. <https://github.com/Blockstream/borromean_paper/blob/master/borromean_draft_0.01_9ade1e49.pdf>

[5] "Borromean Rings". Wolfram MathWorld. <https://mathworld.wolfram.com/BorromeanRings.html>

[6] "Anonymous Identification in Ad Hoc Groups". Yevgeniy Dodis, Aggelos Kiayias, Antonio Nicolosi and Victor Shoup. 2004. <https://link.springer.com/chapter/10.1007/978-3-540-24676-3_36>

[7] "Regarding application of ring signatures". *Maeher*. January 14, 2021. <https://crypto.stackexchange.com/a/87530>

[8] "Traceable Ring Signature". Eiichiro Fujisaki and Koutarou Suzuki. 2007. <https://link.springer.com/chapter/10.1007/978-3-540-71677-8_13>

the two signatures are linked. If the signer signs different messages with the same tag, then the public key of the signer is revealed. This prevents the signer from signing twice for the same topic.

This type of ring signature is very useful in anonymous voting. The scheme prevents an anonymous voter from voting twice. And if doing so, the voter can be prevented from using their public key again. It can be thought of as a blacklist system, useful for avoiding future DoS attacks.

7.2. Linkable Ring Signatures

Linkable ring signatures are a similar scheme.[9] It uses a key image I derived from the private and public keys of the signer with $I = p_s * h(P_s)$, where p_s is the private key.[10] I is then attached to the signature. The image allows linking different messages without revealing too much information about the private/public key pair used. This type of signature also prevents signing twice, while the identity of the signer is always anonymous.

This property is extremely useful for avoiding double spending in anonymous cryptocurrencies like Monero.[11]

7.3. Threshold Ring Signatures

This scheme is the ring version of regular threshold signatures. A threshold signature is an interactive signing scheme that allows a subset of signers to agree to sign a message. In order to sign, a threshold t of N signers must agree, with $t < N$. The ring version works in a similar way with $t < N <= r$.[12] Threshold ring signatures allow anonymity while preserving the functionality of thresholds.

This is useful for two-factor anonymous signing and anonymous multisig (multisignature) wallets in cryptocurrencies.

8. Conclusion

This report introduces the concept of ring signatures. Furthermore, it also explains the security aspects of the scheme and the improvements on its efficiency. Finally, some applications and variants are described. All in all, ring signatures prove to have several uses while guaranteeing anonymity.

[9] “Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups”. Joseph K. Liu, Victor K. Wei and Duncan S. Wong. 2004.
<https://link.springer.com/chapter/10.1007/978-3-540-27800-9_28>

[10] “Zero to Monero”. Koe, Kurt M. Alonso and Sarang Noether. April 4, 2020.
<https://www.getmonero.org/library/Zero-to-Monero-2-0-0.pdf>

[11] "[ANN][BMR] Bitmonero - an anonymous coin based on CryptoNote technology”. *eizh*. April 23, 2014.
<https://bitcointalk.org/index.php?topic=563821.msg6349635#msg6349635>

[12] “Threshold Ring Signatures: New Definitions and Post-quantum Security”. Abida Haque and Alessandra Scafuro. April 29, 2020.
<https://link.springer.com/chapter/10.1007/978-3-030-45388-6_15>