# Physical Ring Signature [⋆]

Xavier Bultel[1][0000−0002−8309−8984]

INSA Centre Val de Loire, Laboratoire d'informatique fondamentale d'Orléans, Bourges, France.
xavier.bultel@insa-cvl.fr

**Abstract.** Ring signatures allow members of a group (called *ring*) to sign a message anonymously within the group, which is chosen ad hoc at the time of signing (the members do not need to have interacted before). In this paper, we propose a physical version of ring signatures. Our signature is based on one-out-of-many signatures, a method used in many real cryptographic ring signatures. It consists of boxes containing coins locked with padlocks that can only be opened by a particular group member. To sign a message, a group member shakes the boxes of the other members of the group so that the coins are in a random state ("heads" or "tails", corresponding to bits 0 and 1), and opens their box to arrange the coins so that the exclusive "or" of the coins corresponds to the bits of the message they wish to sign. We present a prototype that can be used with coins, or with dice for messages encoded in larger (non-binary) alphabets. We suggest that this system can be used to explain ring signatures to the general public in a fun way. Finally, we propose a *semi-formal* analysis of the security of our signature based on real cryptographic security proofs.

**Keywords:** Physical Cryptography · Ring Signature · Anonymity.

## 1 Introduction

The signature is a fundamental primitive of public key cryptography that allows the owner of a secret key to sign messages in such a way that anyone can verify the signature using a public key. In some cases, it may be useful to allow group members to sign on behalf of the group without revealing their personal identity. A simple solution is to agree a priori on a public and secret key pair within the group, but this solution does not allow the signer to dynamically choose group members without consulting them when signing the message.

In 2001, Rivest, Shamir, and Tauman introduced ring signatures in their seminal paper "How to Leak a Secret" [13]. In this primitive, a user generates a signature on behalf of the group using their own secret key and the public keys of the group members (which the user can select when signing). The term "ring" refers to the method of signing used in this pioneering paper: the signer generates a chain of values depending on the message, using successively the public keys of all the group members except their own to encrypt the values. Then, using their own secret key, the signer decrypts the last value in the chain to append it to the beginning of the chain, thus closing the chain in a ring. To verify the signature, a user verifies that the ring is correct by reproducing the successive encryptions with the group members public keys, but cannot guess with which key the ring has been closed.

Ring signatures have always been of great interest because of their relevance to real-world problems, both technical and societal. The first motivation for ring signatures is to protect whistleblowers [13]. For instance, an employee of a company with illegal practices could, if each employee had a public key, expose those practices by signing as a member of the company, but without revealing their exact identity. Less directly, ring signatures have been used in many protocols, such as e-voting and e-cash [17], to guarantee anonymous membership. More recently, they have been used to anonymize certain actions on the blockchain [15], and to prevent transactions in the Monero cryptocurrency from being traced [16].

[⋆] This work has been accepted for publication in the proceedings of the *12th International Conference on Fun with Algorithms* (FUN 2024). A (heavier) version of this report with high resolution images is available at http://www.bultel.info/fun24.pdf.

This primitive is therefore fundamental to many tools designed for a wide audience without advanced computer or mathematical skills. We believe that the use of security technology is only possible if users trust it, which is only possible if they feel that they have understood how their data is being processed and protected. It is thus necessary to find straightforward and convincing ways to explain the mechanisms used in ring signatures in order to inform and reassure the people who use its applications.

In this paper we propose a physical ring signature construction based on everyday objects such as boxes, padlocks, coins, dice, and glass. The actions to be performed are simple and consist of opening and closing the padlocks on the boxes, shaking the boxes, or looking through the glass to see the value of the coins and dice. Some basic calculations are also required, such as adding small integers, which can be done with a calculator. The overall look of the device is intriguing, and it is fun and easy enough for children to use. We believe that this playful aspect makes it an accessible tool for popularising the concept of ring signatures to the general public.

*Technical Overview of our Contributions.* Our signature mechanism is based on one-out-of-many signatures [1], which are themselves inspired by proofs of partial knowledge [6]. This general paradigm has been widely used to construct ring signatures, so the mechanism of our physical ring signature gives an accurate idea of how cryptographic ring signatures are actually designed. In a nutshell, this paradigm is based on message-randomizable signatures, *i.e.*, signature schemes where it is possible to construct signatures on random messages even without knowing the signer's secret key (on the other hand, it is impossible to construct a signature on a fixed message). To sign a message within a group, the signer creates signatures on random values for the public key of all the other members of the group. The signer then computes the bitwise exclusive "or" of all the random values and the message, and signs the result with their secret key. The ring signature is the set of signed values. To verify it, the verifier checks the signature of each value with the public key of the corresponding group member and verifies that the bitwise exclusive "or" of all the values is equal to the message. This method is based on the indistinguishability of the signature made with the secret key from those generated for random values, and on the impossibility of stumbling upon random values that will give the message if no secret key is known.

The first building block in our construction is a physical message-randomizable signature. To do this, we use compartmentalized boxes with a transparent top and place a coin with two different sides in each compartment. The signer manually signs padlocks to which they have the key and locks the boxes with these padlocks. These locked boxes are their public keys, and they distribute them to everyone. To sign a binary message, the signer takes one of their boxes, opens it with their key, and arranges the coins so that they correspond to the bits in the message (we assign "heads" to 0 and "tails" to 1). Note that this operation requires their key. They then close the box. To verify the signature, the verifier checks the manual signature on the padlock and checks that the value of the coins matches the message by looking through the transparent top (without opening the box). To obtain a signature on a random message, anyone can take a box from the signer and shake it so that the coins are disturbed and end up in random "heads" or "tails" state.

The general construction of the physical ring signature results from using the one-out-of-many signatures method applied to our physical message-randomisable signature. As it stands, this construction allows a user to generate ring signatures for random messages, which can be problematic in some cases. We propose a countermeasure where the message must start with a given binary string. To sign longer messages, it is possible to replace the coins with dice. Our prototype uses 30-sided dice, which can encode the letters of the alphabet and certain punctuation marks. Finally, we highlight the security assumptions that our material must verify for our signature to be secure, and we prove the security of our signature in a *semi-formal* way using a sequence of games [14]. By semi-formal, we mean that we try to get close to a real computational security proof. This is not entirely possible because our assumptions are physical and not computational, whereas the ring signature security model considers an adversary modelled by a polynomial time Turing machine.

*Related Works.* Many cryptographic tools have been adapted in physical form. For example, several physical zero-knowledge proofs, mainly using cards and envelopes, have been proposed for many logic puzzles [9,4].

The use of cards has also been exploited to build secure multi-party computation protocols [12]. Other works use different tools, such as coin-based secure computation [10] or light cryptography [11], which uses light and shadows for specific secure computation protocols. Physical secure auction protocols have also been proposed [8], one using envelopes and the other a more complex construction using wooden boxes and padlocks. Another example is the construction of threshold access control using padlocks and latches placed in certain configurations [7]. However, to the best of our knowledge, no physical ring signature has ever been proposed, none of the existing physical primitives can be easily adapted to obtain a physical ring signature, and no physical construction uses a mechanism similar to ours (shaking transparent boxes containing coins/dice to randomise data).

## 2 Technical Background

In this section we first define our notations, then recall the definition of ring signatures and their security properties, and finally recall a property about modular additions.

*Notations.* $(x_i)_{i=0}^{n-1}$ (resp. $\{x_i\}_{i=0}^{n-1}$) denotes the vector (resp. set) containing the idexed elements $x_0, x_1, \cdots$, and $x_{n-1}$, and $\mathbb{Z}_n$ denotes the set of integers modulo $n$ (*i.e.*, $\{i\}_{i=0}^{n-1}$). The expression $y \leftarrow x$ denotes the affectation of the value of the variable $x$ to the variable $y$, the expression $y \leftarrow \mathsf{Algo}(x)$ denotes the affectation of the output of the algorithm $\mathsf{Algo}$ on input $x$ to the variable $y$, and the expression $y \overset{\$}{\leftarrow} S$ denotes the affectation of a value chosen in the uniform distribution on a set $S$ to the variable $y$. The acronym p.p.t. in $\lambda$ means *probabilistic polynomial time in* $\lambda$ (when the context is clear, we omit the parameter $\lambda$).

**Definition 1 (Ring Signature [2]).** *Let $\lambda$ be a security parameter. A ring signature is a tuple of p.p.t. algorithms* $(\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ *defined as follows:*

$\mathsf{Gen}(\lambda)$**:** *on input $\lambda$, returns a pair of public/secret keys* $(\mathsf{pk}, \mathsf{sk})$.

$\mathsf{Sig}(\mathsf{sk}, R, m)$**:** *on input a secret key* $\mathsf{sk}$*, a set of public keys $R$ (containing the public key corresponding to* $\mathsf{sk}$*), and a message $m$, returns a signature $\sigma$.*

$\mathsf{Ver}(R, m, \sigma)$**:** *on input a set of public keys $R$, a message $m$, and a signature $\sigma$, returns a bit $b \in \{$accept, reject$\}$.*

*Moreover, for any integers $s$ and $j$ such that $j < s$, any message $m$, any* $(\mathsf{pk}_i, \mathsf{sk}_i)$ *outputted by* $\mathsf{Gen}(\lambda)$ *for all $i \in \mathbb{Z}_s$, and any $\sigma$ outputted by* $\mathsf{Sig}(\mathsf{sk}_j, \{\mathsf{pk}_i\}_{i=0}^{s-1}, m)$*, the condition* $\mathsf{Ver}(\{\mathsf{pk}_i\}_{i=0}^{s-1}, m, \sigma) =$ accept *is required to hold.*

A secure ring signature is required to satisfy two security properties: unforgeability and anonymity [2]. These properties are modelled by experiments that simulate the use of a ring signature and where a p.p.t. adversary tries to perform an attack.

Unforgeability ensures that a user who is not a member of the group cannot generate a valid signature for a fresh message, even if they can access ring signatures for other messages. In the unforgeability experiment, the adversary is given public keys (corresponding to secret keys they do not know), and can query an oracle for signatures on selected messages using these keys. Their goal is to generate a fresh valid ring signature that has not been generated by the oracle. A ring signature is considered to be unforgeable if no adversary can succeed in this attack with a significant (non-negligible[1]) probability.

**Definition 2 (Unforgeability [2]).** *Let $\lambda$ be a security parameter, let* $\mathsf{RS} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ *be a ring signature, and let $\mathcal{A}$ be a p.p.t. algorithm. For any integer $s$, we define the $s$-unforgeability experiment on* $\mathsf{RS}$ *for $\mathcal{A}$ as follows:*

- *The experiment generates $s$ key pairs* $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=0}^{s-1}$ *and sends* $\{\mathsf{pk}_i\}_{i=0}^{s-1}$ *to $\mathcal{A}$.*
- *$\mathcal{A}$ has access to an oracle* $\mathsf{Sig}(\cdot,\cdot,\cdot)$ *that returns a signature generated by* $\mathsf{Sig}(\mathsf{sk}_j, R, m)$ *on query $(j, R, m)$.*

[1] A function $f$ is negligible in $x$ if for any positive polynomial $p$, there exists un integer $x_0$ such that for all $x > x_0$, $|f(x)| \leq 1/p(x)$

- *$\mathcal{A}$ returns $(R_*, m_*, \sigma_*)$. The experiment returns* 1 *iff* $\mathsf{Ver}(R_*, m_*, \sigma_*) = 1$*,* $R_* \subseteq \{\mathsf{pk}_i\}_{i=0}^{s-1}$*, and no query* $(j, R, m)$ *satisfies* $(R, m) = (R_*, m_*)$*.*

$\mathsf{RS}$ *is said to be unforgeable if for all $s$ and all p.p.t. algorithms $\mathcal{A}$, the probability that the $s$-unforgeability experiment returns* 1 *is negligible in* $\lambda$*.*

Anonymity ensures that it is not possible to guess which member of the group is the author of a given signature. In the anonymity experiment, the adversary is given public/secret keys, chooses two of them, and is given a ring signature generated from one of these two secret keys (and whose ring contains the two public keys). The adversary tries to distinguish which of the two keys was used with a non-negligible advantage.

**Definition 3 (Anonymity [2]).** *Let $\lambda$ be a security parameter, let $\mathsf{RS} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ be a ring signature, and let $\mathcal{A}$ be a p.p.t. algorithm. For any integer $s$ and any bit $b$, we define the $(s, b)$-anonymity experiment on $\mathsf{RS}$ for $\mathcal{A}$ as follows:*

- *The experiment generates $s$ key pairs $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=0}^{s-1}$ and sends $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=0}^{s-1}$ to $\mathcal{A}$.*
- *$\mathcal{A}$ sends $(R, m, i_0, i_1)$ to the experiment. If $R \subseteq \{\mathsf{pk}_i\}_{i=0}^{s-1}$ and $(\mathsf{pk}_{i_0}, \mathsf{pk}_{i_1}) \in R^2$, then the experiment computes $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}_{i_b}, R, m)$, and sends $\sigma$ to $\mathcal{A}$.*
- *$\mathcal{A}$ returns a bit $b_*$.*

$\mathsf{RS}$ *is said to be anonymous if for all $s$ and all p.p.t. algorithms $\mathcal{A}$, the probability that $\mathcal{A}$ returns* 1 *on the $(s, 0)$-anonymity experiment is negligibly close (in $\lambda$) to the probability that $\mathcal{A}$ returns* 1 *on the $(s, 1)$-anonymity experiment.*

The one-out-of-many signatures paradigm [1] presented in Section 1 uses the following result: for any $m \in \mathbb{Z}_n$, if we randomly generate $s$ integers $(x_i)_{i=0}^{s-1}$ whose sum modulo $n$ is $m$ by choosing $j \in \mathbb{Z}_s$, by randomly drawing $x_i \xleftarrow{\$} \mathbb{Z}_n$ for all $i \neq j$, and by completing with the only possible $x_j$, the integers $(x_i)_{i=0}^{s-1}$ and $m$ do not reveal any information about $j$. For instance, for $m = 0$, $n = 2$ and $s = 2$, if we randomly draw $x_0$ then we should set $x_1 = x_0$ to get $x_0 + x_1 \bmod 2 = 0$, and if we randomly draw $x_1$ then we should set $x_0 = x_1$; both cases return $(x_0, x_1) = (0, 0)$ and $(x_0, x_1) = (1, 1)$ with the same probability. On the other hand, for $m = 1$, the two cases $(0, 1)$ and $(1, 0)$ have the same probability, no matter which element was randomly generated. This result is generalised for vectors of integers in the following theorem. A proof of this theorem is given in Appendix A.

**Theorem 1.** *Let $N, n$, and $s$ be three integers. For any $m \in \mathbb{Z}_n^N$, any pair $(i_0, i_1) \in \mathbb{Z}_s^2$, and any distinguisher $\mathcal{D}$, we have:*

$$\Pr\left[\begin{array}{l}\forall i \in \mathbb{Z}_s \backslash \{i_0\}, x_i \xleftarrow{\$} \mathbb{Z}_n^N; \\ \forall j \in \mathbb{Z}_N, x_{i_0,j} \leftarrow \left(m_j - \sum_{i=0; i \neq i_0}^{s-1} x_{i,j}\right) \bmod n;\end{array} : 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s-1})\right] =$$
$$\Pr\left[\begin{array}{l}\forall i \in \mathbb{Z}_s \backslash \{i_1\}, x_i \xleftarrow{\$} \mathbb{Z}_n^N; \\ \forall j \in \mathbb{Z}_N, x_{i_1,j} \leftarrow \left(m_j - \sum_{i=0; i \neq i_1}^{s-1} x_{i,j}\right) \bmod n;\end{array} : 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s-1})\right]$$

An example of unforgeable and anonymous cryptographic ring signature based on the BLS [3] signature that follows the one-out-of-many signatures paradigm [1] is given in [5].

## 3 Our Physical Ring Signature

In this section we present our physical ring signature scheme. We first introduce the material required, then explain how to use it to design a physical message-randomisable signature, and finally explain how a user can anonymously sign within a group using it. We illustrate the steps involved with the help of a physical prototype that we have built.

### 3.1 Material

Each member of the ring/group must be provided with indelible felt-tip pens to enable them to make indelible manual signatures on any surface. Each user must also have an unlimited number of padlocks. Padlocks belonging to the same user must be identical and have a single key that can be used to open them. We assume that users have access to an unlimited number of coins whose two sides (heads and tails) are easily distinguishable. We associate "heads" with the binary value 0 and "tails" with the binary value 1. The coins and the padlocks we use for our prototype are shown in Figure 1.

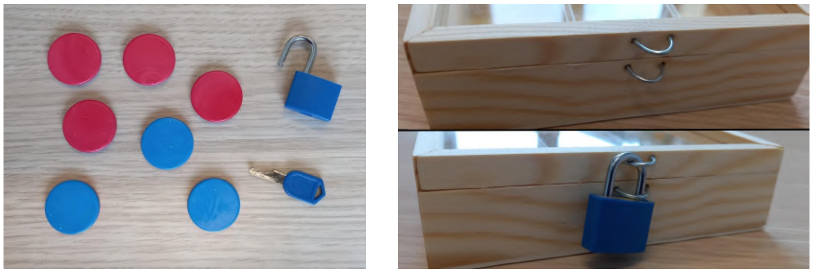

**Fig. 1.** On the left, some coins with red "heads" and blue "tails" associated with the values 0 and 1 and a padlock with its key, and on the right, a latch on a box, with and without the padlock.

An unlimited number of compartmentalised boxes are also available (see Figure 2), with the following features:

- The box has a lid that can be opened by a mechanism.
- Each box is divided into $N$ compartments indexed from 0 to $N-1$ so that it is not possible to move an object from one compartment to another when the box is closed.
- When the box is open, a user can freely place or remove objects in each compartment.
- Each compartment has the shape of a parallelepiped whose edges are larger than the diameter of the coins. A coin in a compartment can therefore move freely within the space of the compartment.
- The lid of the box is transparent so that the top of an object in a compartment can be clearly seen.
- The box has a latch that prevents it from being opened (see Figure 1). This latch must be perfectly lockable with a padlock, preventing anyone from opening the box without the padlock key.

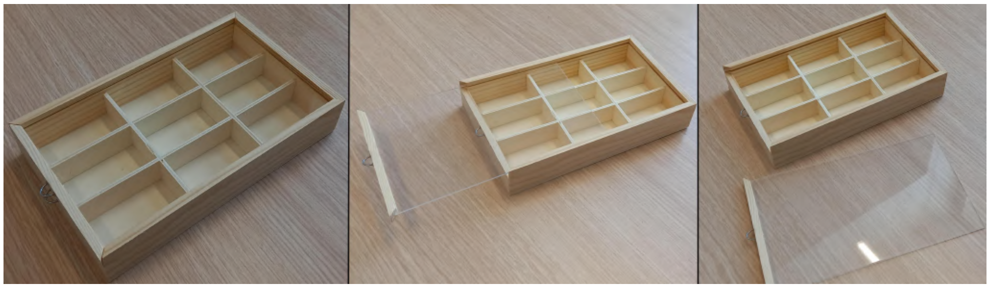

**Fig. 2.** A box with 9 compartments and with a slip-on lid.

As explained in Section 1, our ring signature is based on the one-out-of-many signatures paradigm, which uses message-randomisable signatures. Therefore, we first present how to generate physical message-randomisable signatures, before showing how to use them to generate our physical ring signature.

### 3.2 Key Generation

We have assumed that a user has an unlimited number of identical padlocks and a single key that can be used to open these padlocks. The user manually signs their padlocks with an indelible felt-tip pen, so that the signature is visible and can be verified by anyone. The user then fills the compartments of several boxes by placing a coin in each compartment. The visible side of the coin ("heads" or "tails") in each compartment is randomly chosen. The user closes each of these boxes with one of their signed padlocks. These boxes containing coins and closed by signed padlocks are their public key $\mathsf{pk}$ (see Figure 3) and they distribute them to the other users. If necessary, they can create new ones at any time. Their secret key $\mathsf{sk}$ is the key that opens their padlocks.

### 3.3 Message-randomisable Signature Generation and Verification

To **sign** a binary message $(m_i)_{i=0}^{N-1}$ of $N$ bits, the signer takes one of their public keys (*i.e.*, a box divided into compartments numbered from 0 to $N-1$ containing coins and closed by a signed padlock for which they have the key), uses their key to open the padlock and the box to arrange the coins so that the state of the coin in the $i$-th compartment ("heads" or "tails") corresponds to the $i$-th bit $m_i$ of the message, and then closes the padlock on the box.

To **verify** this signature, the verifier checks that the padlock in the box has been manually signed by the signer (by comparing the manual signature with the manual signature on the padlock of one of the signer's public keys), and looks through the transparent lid to check that the state of the coins corresponds to the bits of the message $(m_i)_{i=0}^{N-1}$.

Without knowing the secret key (and therefore without being able to open the box), any user can **generate the signature for a random message** by shaking the box: since the diameter of the coins is smaller than the dimensions of the compartments, the coins can turn on themselves and randomly land on one of their sides, forming a random binary message.

### 3.4 Ring Signature Generation

We now show how to use the physical message-randomizable signatures to create physical ring signatures. Let $s$ be the size of the ring/group. The signer has all the (indexed) public keys $R = \{\mathsf{pk}_i\}_{i=0}^{s-1}$ of the members of the group (including their own), their secret key $\mathsf{sk}$, and wishes to sign a message $m = (m_i)_{i=0}^{N-1}$ of $N$ bits. The index corresponding to their public key is denoted $j$. Recall that public keys are closed boxes containing coins whose visible sides are random. For each $\mathsf{pk}_i$ such that $i \neq j$, the signer shakes the box so that the coins in the compartments are randomly flipped (this is the mechanism used to generate signatures for random messages without knowing the key, as described above).

We set $c_{i,k}$ to the binary value associated with the state of the coin in the $k$-th compartment of the box of the $i$-th public key $\mathsf{pk}_i$. Using their secret padlock key, the signer opens the box corresponding to their public

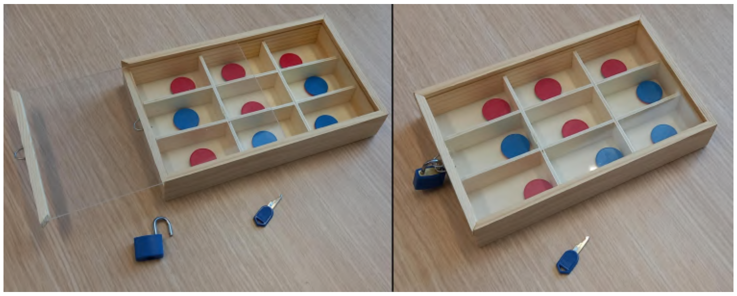

**Fig. 3.** A padlocked box (opened then closed) corresponding to a user's public key, and the padlock key corresponding to their secret key.

key $\mathsf{pk}_j$ and manually arranges the coins (this is the mechanism used to sign a given message by knowing the key described above) so that $\sum_{i=0}^{s-1} c_{i,k} \bmod 2 = m_k$.

To put it in simple terms, this operation is equivalent to the following: for each bit $m_i$ of the message, if $m_i = 0$ (resp. $m_i = 1$), then the signer places the coin in the $i$-th compartment of their own box, so that there is an even (resp. odd) number of coins on "tails" (corresponding to the bit 1) in the $i$-th compartments of all the boxes of all members.

The signer then closes the padlock on their box and arranges the boxes in random order. The signature is the set of all the boxes of the members after these operations.

### 3.5 Signature Verification

The verifier receives the signature that consists on $s$ padlocked boxes signed by the $s$ members of the group, and the binary message $(m_i)_{i=0}^{N-1}$. They first check that the padlocks have all been manually signed by a different member of the group, and that these signatures are valid (by comparing it with the manual signatures on the padlocks of the group member's public keys). The verifier sets $c_{i,k}$ to the binary value associated with the state of the coin in the $k$-th compartment of the box signed by the $i$-th member of the group. They check that $\sum_{i=0}^{N-1} c_{i,k} \bmod 2 = m_k$ for each $k$.

To put it in simple terms, this operation is equivalent to the following: for each bit $m_i$ of the message, if $m_i = 0$ (resp. $m_i = 1$), then the verifier checks that there is an even (resp. odd) number of coins on "tails" (corresponding to the bit 1) in the $i$-th compartments of all the boxes of all members. If this is the case, they accept the signature, if not, they refuse it.

### 3.6 Preventing the Signing of Random Messages

As it stands, it is possible to generate signatures on random messages without having any padlock key: all you have to do is take the group member boxes and shake them. The result is a signed message whose bits are the modulo two additions of the random values associated to the state of the coins in each compartment. To prevent this, the following countermeasure can be applied: given a security parameter $\lambda$, the first $\lambda$ bits of the signed message must be 0, otherwise the signature is not valid. Thus, the probability of obtaining a valid signature by shaking the boxes is $1/2^\lambda$, which is negligible in $\lambda$. On the other hand, the size of the actual signed message becomes $N - \lambda$ bits.

### 3.7 Example Using a Prototype

We have built a prototype of our signature using boxes with $N = 9$ compartments. In Figure 4, we show all the steps to generate a ring signature of a message $M = 111111$ with $\lambda = 3$ and $s = 3$ group members. Each group member has a different coloured padlock (blue, orange, and pink) and we have omitted the manual signatures on the padlocks. The signer is the owner of the key for the pink padlock. We set $m = (m_k)_{k=0}^{N-1} = (0, 0, 0, 1, 1, 1, 1, 1, 1)$ the message that starts with $\lambda = 3$ times 0 and ends with the bits of $M$.

**Step A:** The signer places the three public key boxes side by side. The state of the coins in each of these boxes is associated with the respective values:

$$c_B = (c_{B,k})_{k=0}^{N-1} = (1, 1, 0, 0, 1, 0, 1, 1, 0)$$
$$c_O = (c_{O,k})_{k=0}^{N-1} = (0, 1, 1, 1, 0, 1, 0, 1, 1)$$
$$c_P = (c_{P,k})_{k=0}^{N-1} = (1, 1, 0, 1, 1, 0, 1, 0, 0)$$

**Step B:** The signer shakes the blue member box.
**Step C:** The signer shakes the orange member box.

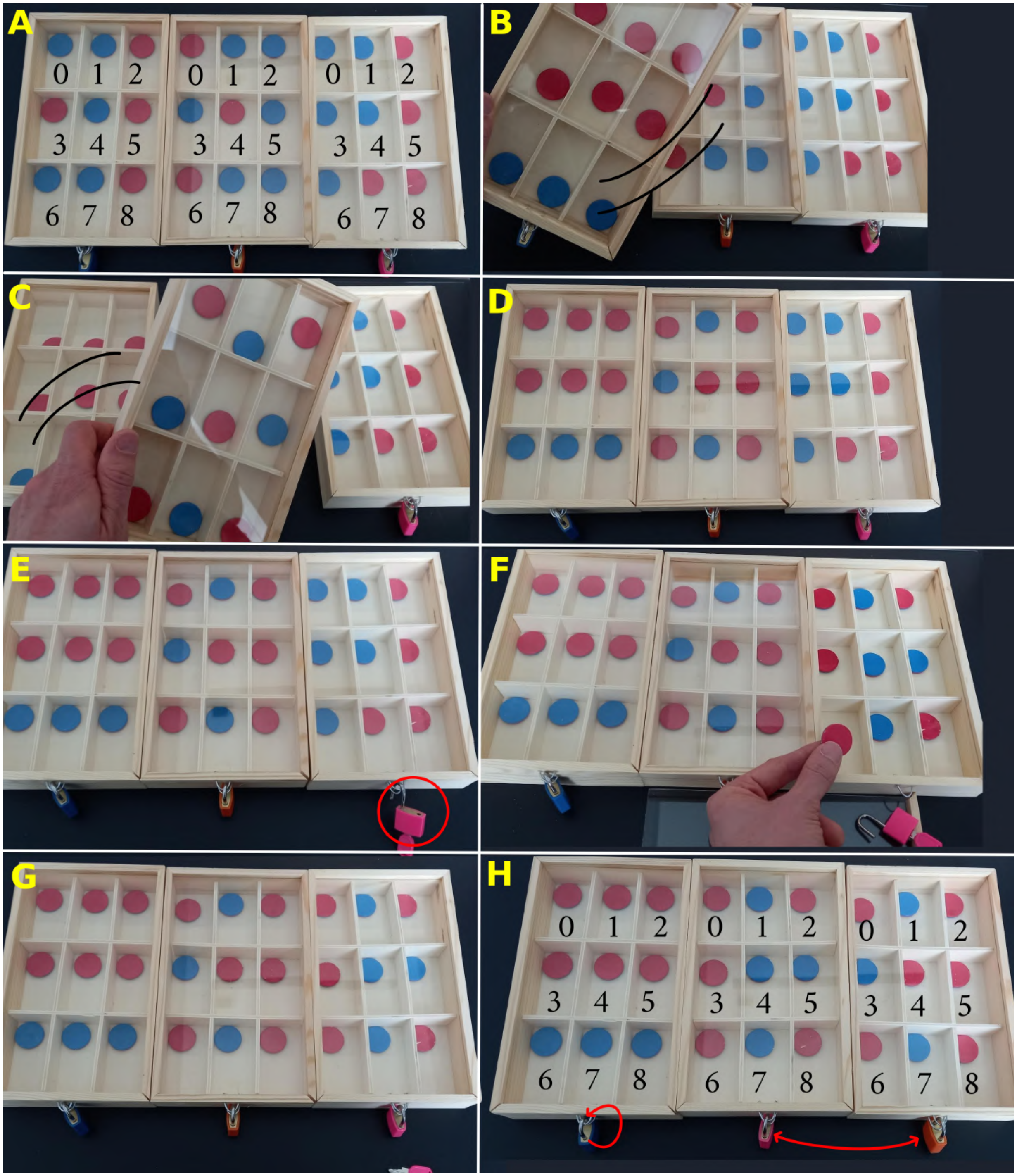

**Fig. 4.** Signature of $M = 111111$ in a ring of $s = 3$ users with security parameter $\lambda = 3$ and boxes with $N = 9$ compartments.

**Step D:** At this step, the coins in each box are associated with the values:

$$\begin{aligned} c_B &= (c_{B,k})_{k=0}^{N-1} = (0,0,0,0,0,0,1,1,1) \\ c_O &= (c_{O,k})_{k=0}^{N-1} = (0,1,0,1,0,0,0,1,0) \\ c_P &= (c_{P,k})_{k=0}^{N-1} = (1,1,0,1,1,0,1,0,0) \end{aligned}$$

**Step E:** The signer opens the pink paddlock using their key, then opens the box.

**Step F:** The signer rearranges the coins in their open box in such a way that $c_{P,k} = m_k \oplus c_{B,k} \oplus c_{O,k}$ for $0 \leq k < N$ (exclusive "or" $\oplus$ is equivalent to addition/substraction modulo 2). In other words, for each index $k$ they make the number of coins on the blue side (corresponding to 1) even if the bit of the message $m_k$ is 0, and odd if the bit of the message $m_k$ is 1. This results in the following configuration:

$$\begin{aligned} c_B &= (c_{B,k})_{k=0}^{N-1} = (0,0,0,0,0,0,1,1,1) \\ c_O &= (c_{O,k})_{k=0}^{N-1} = (0,1,0,1,0,0,0,1,0) \\ c_P &= (c_{P,k})_{k=0}^{N-1} = (0,1,0,0,1,1,0,1,0) \end{aligned}$$

**Step G:** The signer closes their box with their padlock.

**Step H:** The signer shuffles the boxes: the blue member's box stays first, and the orange and pink members' boxes are swapped.

At the end of the signature, the coins are associated with the following binary values:

$$\begin{aligned} c_B &= (c_{B,k})_{k=0}^{N-1} = (0,0,0,0,0,0,1,1,1) \\ c_P &= (c_{P,k})_{k=0}^{N-1} = (0,1,0,0,1,1,0,1,0) \\ c_O &= (c_{O,k})_{k=0}^{N-1} = (0,1,0,1,0,0,0,1,0) \end{aligned}$$

Any user can compute: $(c_{B,k} \oplus c_{P,k} \oplus c_{O,k})_{k=0}^{N-1} = (0 \oplus 0 \oplus 0, 0 \oplus 1 \oplus 1, 0 \oplus 0 \oplus 0, 0 \oplus 0 \oplus 1, 0 \oplus 1 \oplus 0, 0 \oplus 1 \oplus 0, 1 \oplus 0 \oplus 0, 1 \oplus 1 \oplus, 1 \oplus 0 \oplus 0) = (0,0,0,1,1,1,1,1,1)$, and thus verify that $(c_{B,k} \oplus c_{P,k} \oplus c_{O,k})_{k=0}^{\lambda-1} = (0,0,0)$ and $(c_{B,k} \oplus c_{P,k} \oplus c_{O,k})_{k=\lambda}^{N-1} = M$. This is equivalent to verifying that for each index $k$, the number of blue coins in the $k$-th compartments of the 3 boxes is even if $m_k = 0$ (*i.e.*, for $0 \leq k \leq 2$), and odd if $m_k = 1$ (*i.e.*, for $3 \leq k \leq 8$). Note that the probability of producing a valid signature without the keys (by shaking the boxes only) is $1/2^\lambda = 1/8$ (this is the probability that the first three sums of bits give 0). Of course, to have a more realistic probability of preventing the generation of random message signatures, we would need to use boxes with more compartments.

### 3.8 Generalising on Larger Alphabets with Dice

A coin can be thought of as a two-sided die. By generalising the principle of our signature, we could sign messages on alphabets of $n$ symbols using $n$-sided dice (numbered from 1 to $n$). For example, using 30-sided dice, any integer $i$ between 1 and 26 can be associated with the $i$-th letter of the Latin alphabet, and $27, 28, 29$, and 30 can be respectively associated with space, comma, dot, and a special character '`*`'. Figure 5 shows our prototype used with 30-sided dice (whose diameter is small enough for the dice to roll freely through the compartments).

The idea remains the same: the $i$-th character of the signed message corresponds to the sum modulo $n$ of the values at the top of the dice in the $i$-th compartment of the boxes of the members of the group. The first $\lambda$ characters must correspond to the special character '`*`', so the probability of generating a signature for a valid random message is $1/n^\lambda$. On the other hand, a signer who has the key to one of the padlocks will always be able to arrange the dice to obtain a valid signature for a given message.

In Figure 6, we show all the steps to generate a ring signature of a message $M =$ `"hello."` with our prototype using 30-sided dice and with $\lambda = 3$ and $s = 3$. Each user has a different coloured padlock (green, red, and yellow) and we have omitted the manual signatures on the padlocks. The signer is the owner of the

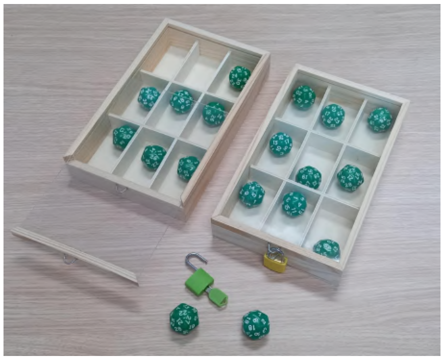

**Fig. 5.** Two public keys of our prototype used with 30-sided dice.

key for the yellow padlock. We set $m = (m_k)_{k=0}^{N-1} = (0, 0, 0, 8, 5, 12, 12, 15, 29)$ the message that starts with $\lambda = 3$ times 0 (that corresponds to the sepcial character '`*`' since $30 \bmod 30 = 0$) and ends with the integers that correspond to the characters '`h`', '`e`', '`l`', '`l`', '`o`', and '`.`'.

**Step A:** The signer places the three public key boxes side by side. The dice in each of these boxes indicate the respective values:

$$\begin{aligned} c_G = (c_{G,k})_{k=0}^{N-1} &= (28, 11, 29, 13, 25, 28, 30, 15, 11) \\ c_R = (c_{R,k})_{k=0}^{N-1} &= (9, 11, 9, 16, 27, 30, 13, 1, 8) \\ c_Y = (c_{Y,k})_{k=0}^{N-1} &= (21, 8, 21, 15, 27, 1, 16, 15, 20) \end{aligned}$$

**Step B:** The signer shakes the green group member box.
**Step C:** The signer shakes the red group member box.
**Step D:** At this step, the dice in each of these boxes indicate the respective values:

$$\begin{aligned} c_G = (c_{G,k})_{k=0}^{N-1} &= (6, 1, 29, 13, 28, 20, 8, 27, 11) \\ c_R = (c_{R,k})_{k=0}^{N-1} &= (7, 29, 28, 14, 10, 17, 16, 19, 21) \\ c_Y = (c_{Y,k})_{k=0}^{N-1} &= (21, 8, 21, 15, 27, 1, 16, 15, 20) \end{aligned}$$

**Step E:** The signer opens the yellow paddlock using their key, then opens the box.
**Step F:** The signer rearranges the dice in their open box in such a way that $c_{Y,k} = m_k - c_{G,k} - c_{R,k} \bmod 30$ for $0 \leq k < N$. This results in the following configuration (where 0 is encoded by 30 on the dice, since dice are numbered from 1 to 30):

$$\begin{aligned} c_G = (c_{G,k})_{k=0}^{N-1} =& (6, 1, 29, 13, 28, 20, 8, 27, 11) \\ c_R = (c_{R,k})_{k=0}^{N-1} =& (7, 29, 28, 14, 10, 17, 16, 19, 21) \\ c_Y = (c_{Y,k})_{k=0}^{N-1} =& (-6 - 7 \bmod 30, -1 - 29 \bmod 30, -29 - 28 \bmod 30, \\ & 8 - 13 - 14 \bmod 30, 5 - 28 - 10 \bmod 30, 12 - 20 - 17 \bmod 30, \\ & 12 - 8 - 16 \bmod 30, 15 - 27 - 19 \bmod 30, 29 - 11 - 21 \bmod 30) \\ =& (17, 0, 3, 11, 27, 5, 18, 29, 27) \end{aligned}$$

**Step G:** The signer closes their box with their padlock.
**Step H:** The signer shuffles the boxes: the order of the boxes changes from green then red then yellow to red then yellow then green.

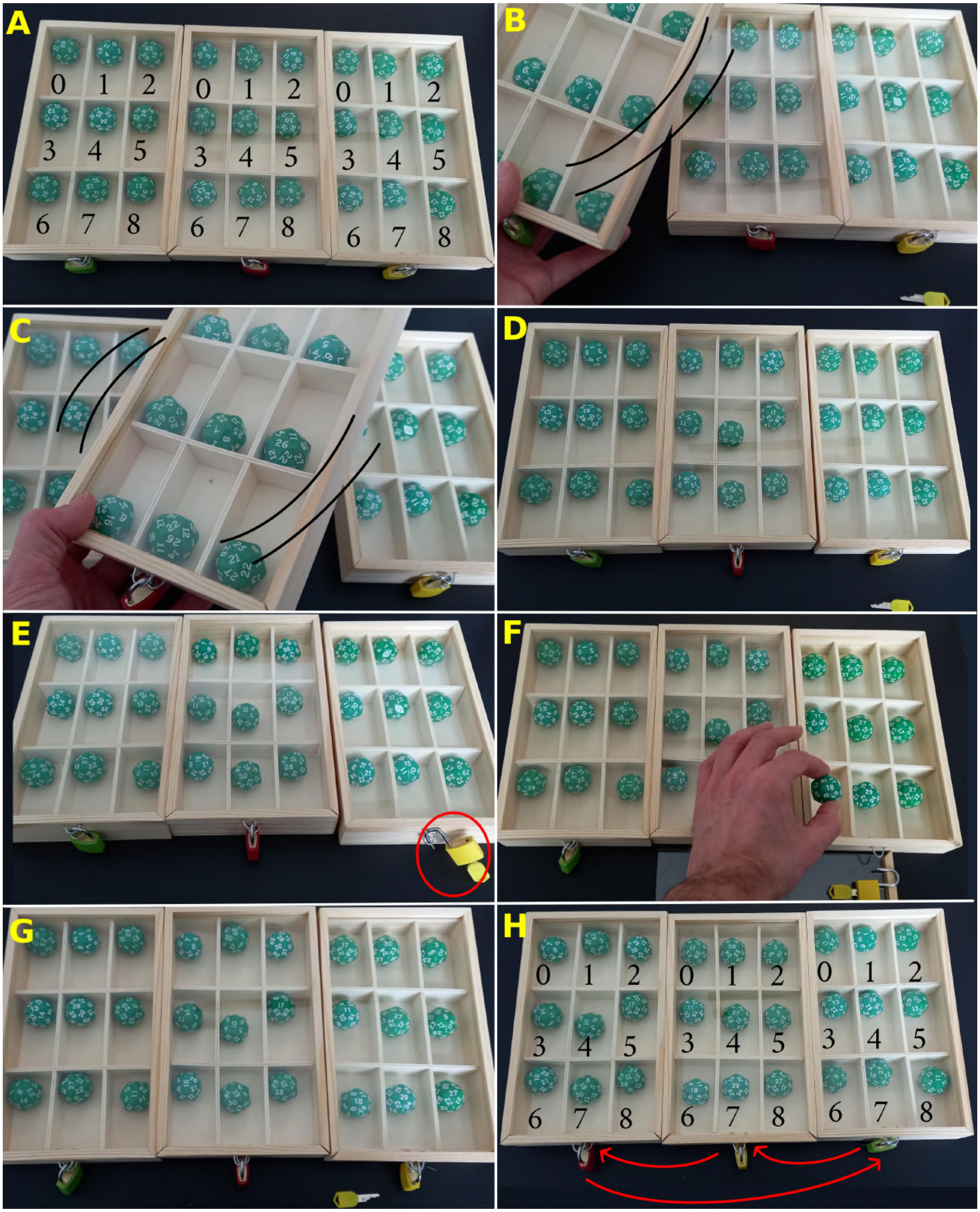

**Fig. 6.** Signature of $M =$ `"hello."` in a ring of $s = 3$ users with security parameter $\lambda = 3$ and boxes with $N = 9$ compartments. See `http://www.bultel.info/prs_30.png` for a higher resolution version of this picture.

At the end of the signature, the dice are in the following configuration:

$$\begin{aligned}c_R &= (c_{R,k})_{k=0}^{N-1} = (7, 29, 28, 14, 10, 17, 16, 19, 21)\\ c_Y &= (c_{Y,k})_{k=0}^{N-1} = (17, 0, 3, 11, 27, 5, 18, 29, 27)\\ c_G &= (c_{G,k})_{k=0}^{N-1} = (6, 1, 29, 13, 28, 20, 8, 27, 11)\end{aligned}$$

Any user can verify the signature by computing:

$$\begin{aligned}&(c_{R,k} + c_{Y,k} + c_{G,k} \ \mathsf{mod}\ 30)_{k=0}^{N-1}\\ &= (7 + 17 + 6 \ \mathsf{mod}\ 30, 29 + 0 + 1 \ \mathsf{mod}\ 30, 28 + 3 + 29 \ \mathsf{mod}\ 30,\\ &\ 14 + 11 + 13 \ \mathsf{mod}\ 30, 10 + 27 + 28 \ \mathsf{mod}\ 30, 17 + 5 + 20 \ \mathsf{mod}\ 30,\\ &\ 16 + 18 + 8 \ \mathsf{mod}\ 30, 10 + 29 + 27 \ \mathsf{mod}\ 30, 21 + 27 + 11 \ \mathsf{mod}\ 30)\\ &= (0, 0, 0, 8, 5, 12, 12, 15, 29),\end{aligned}$$

and thus verify that $(c_{R,k} + c_{Y,k} + c_{G,k} \ \mathsf{mod}\ 30)_{k=0}^{\lambda-1} = (0, 0, 0)$ and that $(c_{R,k} + c_{Y,k} + c_{G,k} \ \mathsf{mod}\ 30)_{k=\lambda}^{N-1} = M$.

Note that the probability of producing a valid signature without the keys (by shaking the boxes only) is $1/n^\lambda = 1/27000$ (this is the probability that the first three sums give 0).

## 4 Security Analysis

In this section, we identify the assumptions required to ensure the security of our prototype.

**Assumptions.** The following properties are assumed to be true:

1. It is not possible to break or force open a compartmentalised box, *i.e.*, a box can only be opened by its lid using the mechanism provided for this purpose.
2. It is impossible to forge a user's manual signature without being that user.
3. Padlocks are unbreakable and cannot be opened without a key. In particular, it is impossible to forge a key for a padlock.
4. A padlock attached to the latch of a box prevents the lid of the box from being opened by the mechanism provided for this purpose.
5. An object cannot be moved to another compartment when a box is locked.
6. The only action that can be performed on a closed box to move the objects contained in its compartments is to shake it.
7. Shaking the box when it contains dice is equivalent to rolling the dice. More precisely, given a box with dice, and knowing the previous position of the dice, it is impossible to determine whether the box has been shaken, or opened, the dice rearranged to show a random value chosen from the uniform distribution, and the box closed again.

These assumptions allow us to claim the following two theorems. For each of them we give an intuitive explanation of the results, then we give a security proof in a *semi-formal* style: we show a sequence of games [14] that reduce to each other by hops, eliminating events whose probability of occurrence is at most negligible under our assumptions. In our security proofs, we consider that the physical actions of closing a padlock, opening a padlock, closing a box, opening a box, moving an object, and shaking a box are achievable by an adversary (modelled by a p.p.t. algorithm) in constant time. In general, no basic physical operation depends on the security parameter $\lambda$. We also consider that any physical action performed by the adversary can be observed by the challenger who simulates the security experiment for them.

**Theorem 2.** *Our physical ring signature is unforgeable under Assumption 1, 2, 3, 4, 5, 6, and 7. More precisely, for any integer s, any security parameter $\lambda$, any p.p.t. algorithm $\mathcal{A}$, and any polynomial function q, the probability that the s-unforgeability experiment on our physical ring signature using n-sided dice for $\mathcal{A}$ returns* 1 *is bounded by $\frac{q(\lambda)}{n^\lambda}$, where $q(\lambda)$ is the number of times that a box is shaken during the experiment.*

According to Assumptions 1, 2, 3, and 4, an adversary cannot manually sign a box in place of a member of the group, and cannot break/force the box and its mechanism if it is locked by a padlock. Nor can they open a box to manually change the value of the dice under Assumption 5 and 6. Furthermore, according to Assumption 6 and 7, they cannot bias the roll of the dice so that it is not uniform when they decide to shake a box. Their only possible strategy is to hope that shaking the boxes will produce a valid signature. To do this, the first $\lambda$ dice in the boxes must match the special character '$*$', which happens with a negligible probability of at most $\frac{q(\lambda)}{n^\lambda}$ where $q(\lambda)$ is the number of times a box is shaken.

*Proof (Proof (Theorem 2)).* Let $s$ be an integer, and $\mathcal{A}$ be a p.p.t. algorithm. We consider the following sequence of games.

Game $G_0$: In this game, a challenger simulates the $s$-unforgeability experiment on our physical ring signature for $\mathcal{A}$. The event "$\mathcal{A}$ wins $G_0$" denotes that the unfogeability expermient returns 1.

Game $G_1$: Same as $G_0$, except that if $\mathcal{A}$ breaks or opens a box on an other way that by its lid using the mechanism provided for this purpose, forge a group member's manual signature, or breaks or opens a padlock without its key, then the challenger returns 0. According with Assumption 1, 2, and 3, we have that $\Pr[\mathcal{A} \text{ wins } G_0] = \Pr[\mathcal{A} \text{ wins } G_1]$.

Game $G_2$: Same as $G_1$, except that if $\mathcal{A}$ opens the box in one of the public keys $\mathsf{pk}_i$, then the challenger returns 0. Note that at this step the public key boxes cannot be opened except by their normal opening mechanism, and each box is locked with a padlock that cannot be broken. According with assumption 4, we have that $\Pr[\mathcal{A} \text{ wins } G_1] = \Pr[\mathcal{A} \text{ wins } G_2]$.

Game $G_3$: Same as $G_2$, except that if $\mathcal{A}$ moves one die to another compartment in the box of a public key, then the challenger returns 0. Note that at this step the public key boxes cannot be opened. According with assumption 5, we have that $\Pr[\mathcal{A} \text{ wins } G_2] = \Pr[\mathcal{A} \text{ wins } G_3]$.

At this step, the only way to change the position of the dice in the boxes corresponding to the public keys is to shake them, according to Assumption 6. In addition, according to Assumption 7, shaking a box is equivalent to giving random values to the dice inside.

Game $G_4$: Same as $G_3$, except that if $\mathcal{A}$ returns a valid signature beginning with $\lambda$ times the special character '$*$', then the challenger returns 0. We claim that $|\Pr[\mathcal{A} \text{ wins } G_3] - \Pr[\mathcal{A} \text{ wins } G_4]| \leq \frac{q(\lambda)}{n^\lambda}$.

In order for the $k$-th symbol of the message to be '$*$', we must have the property $\sum_{i=0}^{s-1} c_{i,k} \bmod n = 0$, where $c_{i,k}$ is the value indicated by the die in the $k$-th compartment of the box of the $i$-th member of the group. Since the $c_{i,k}$ are all drawn in a uniform distribution (when the box is shaken), evaluating the probability of having $\sum_{i=0}^{s-1} c_{i,k} \bmod n = 0$ is equivalent to fixing the value $c_{i,k}$ of the last $s-1$ dice and evaluating the probability of drawing the value $c_{0,k}$ verifying $\sum_{i=0}^{s-1} c_{i,k} \bmod n = 0$. This probability is $1/n$. Moreover, the adversary must succeed this for all $0 \leq k \leq \lambda - 1$, knowing that all the $c_{i,k}$ are randomly refreshed when a box is shaken. Thus, the probability of drawing the values $c_{0,k}$ verifying $\sum_{i=0}^{s-1} c_{i,k} \bmod n = 0$ for all $0 \leq k \leq \lambda - 1$ is $1/n^\lambda$. Since the number of times a box is shaken during the experiment is $q(\lambda)$, the adversary has at most $q(\lambda)$ tries to draw the correct $c_{0,k}$ (we assume that the box of each public key was shaken before being distributed). So $\mathcal{A}$ has a probability bounded by $\frac{q(\lambda)}{n^\lambda}$ to draw all the correct $c_{0,k}$ together for $0 \leq k \leq \lambda - 1$ during the experiment, and so having $\lambda$ times the symbol '$*$' at the beginning of its returned signing message, which concludes the proof of the claim.

Note that in $G_4$, $\mathcal{A}$ can no longer win, since a signature must begin with $\lambda$ times the special character '$*$' to be valid, so its probability of winning the game is 0. Since $\Pr[\mathcal{A} \text{ wins } G_0] = \Pr[\mathcal{A} \text{ wins } G_3]$, we have $|\Pr[\mathcal{A} \text{ wins } G_0] - \Pr[\mathcal{A} \text{ wins } G_4]| \leq \frac{q(\lambda)}{n^\lambda}$, so $\Pr[\mathcal{A} \text{ wins } G_0] \leq \frac{q(\lambda)}{n^\lambda}$. Finally, the probability that the unforgeability experiment returns 1 is bounded by the negligible function $\frac{q(\lambda)}{n^\lambda}$, which concludes the proof.

**Theorem 3.** *Our physical ring signature is anonymous under Assumption 7. More precisely, for any integer $s$, any security parameter $\lambda$, and any p.p.t. algorithm $\mathcal{A}$, the probability that $\mathcal{A}$ returns 1 on the $(s, 0)$-anonymity experiment is equals to the probability that $\mathcal{A}$ returns 1 on the $(s, 1)$-anonymity experiment on our physical ring signature.*

Assumption 7 ensures that it is not possible to distinguish from a physical point of view whether a box has been shaken or whether the things in it have been moved manually, and Theorem 1 ensures that it is not possible to distinguish from a computational point of view which dice value have been drawn randomly and which have been chosen to complete the sum in order to obtain the message. Thus, an adversary has no way of distinguishing which box has been opened, and therefore the identity of the signer.

*Proof (Proof (Theorem 3)).* Let $s$ be an integer, and $\mathcal{A}$ be a p.p.t. algorithm. We consider the following sequence of games.

Game $G_0$: In this game, a challenger simulates the $(s, 0)$-anonymity experiment on our physical ring signature for $\mathcal{A}$. The event "$b_* = 1$ in $G_0$" denotes that $\mathcal{A}$ returns 1 at the end of the experiment.

Game $G_1$: Same as $G_0$, except that when the challenger signs the message $m$ chosen by $\mathcal{A}$, each time they are supposed to shake a box, they instead open it, manually arrange the dice to give them a random value, and close the box again. Under Assumption 7, we have that $\Pr[b_* = 1 \text{ in } G_0] = \Pr[b_* = 1 \text{ in } G_1]$.

Game $G_2$: Same as $G_1$, except that the challenger signs with $\mathsf{sk}_{i_1}$ instead of $\mathsf{sk}_{i_0}$. Assuming that $\Pr[b_* = 1 \text{ in } G_1] \neq \Pr[b_* = 1 \text{ in } G_2]$, we will show that there exists a distinguisher $\mathcal{D}$ that contradicts Theorem 1.

We build $\mathcal{D}$ as follows: $\mathcal{D}$ simulates $G_2$ to $\mathcal{A}$, receives $(R, m, i_0, i_1)$ from $\mathcal{A}$, sets $|R| = s'$, and receives the input $(x_i)_{i=0}^{s'-1}$. We parse $R$ as $\{\mathsf{pk}'_i\}_{i=0}^{s'-1}$, $m$ as $(m_k)_{k=0}^{N-1}$, and $x_i$ as $(x_{i,k})_{k=0}^{N-1}$ for $0 \leq i < s'$. According to the definition of $\mathcal{D}$ in Theorem 1 on the values $N, n, s', m$, and $(i_0, i_1)$, the input $(x_i)_{i=0}^{s'-1}$ verifies $m_k = \sum_{i=0}^{s'-1} x_{i,k} \bmod n$ for all $0 \leq k < N$. To forge the signature, $\mathcal{D}$ opens the box corresponding to each key $\mathsf{pk}'_i$ and arranges the dice in such a way that the die in the $k$-th compartment of the $i$-th box indicates the value $x_{i,k}$. Finally, $\mathcal{D}$ sends the resulting signature to $\mathcal{A}$, receives $b*$ from $\mathcal{A}$, and returns it. We have:

$$\Pr\left[\begin{array}{l}\forall i \in \mathbb{Z}_{s'}\backslash\{i_0\}, x_i \xleftarrow{\$} \mathbb{Z}_n^N;\\ \forall k \in \mathbb{Z}_N, x_{i_0,k} \leftarrow \left(m_k - \sum\limits_{i=0;i\neq i_0}^{s-1} x_{i,k}\right) \mathsf{mod}\ n;\end{array} \ : \ 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s'-1})\right]$$
$$= \Pr[b_* = 1 \text{ in } G_1];$$

$$\Pr\left[\begin{array}{l}\forall i \in \mathbb{Z}_{s'}\backslash\{i_1\}, x_i \xleftarrow{\$} \mathbb{Z}_n^N;\\ \forall k \in \mathbb{Z}_N, x_{i_1,k} \leftarrow \left(m_k - \sum\limits_{i=0;i\neq i_1}^{s-1} x_{i,k}\right) \mathsf{mod}\ n;\end{array} \ : \ 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s'-1})\right]$$
$$= \Pr[b_* = 1 \text{ in } G_2].$$

This contradicts Theorem 1 on the values $N, n, s', m$, and $(i_0, i_1)$ for the distinguisher $\mathcal{D}$, because $\Pr[b_* = 1 \text{ in } G_1] \neq \Pr[b_* = 1 \text{ in } G_2]$. Finally, we deduce that $\Pr[b_* = 1 \text{ in } G_1] = \Pr[b_* = 1 \text{ in } G_2]$.

Game $G_3$: Same as $G_2$, except that when the challenger signs the message $m$ chosen by $\mathcal{A}$, each time they are supposed to open a box, manually arrange the dice to give them a random value, and close the box again, they instead shake the box. Under Assumption 7, we have that $\Pr[b_* = 1 \text{ in } G_2] = \Pr[b_* = 1 \text{ in } G_3]$.

We observe that in $G_3$ the challenger simulates the $(s, 1)$-anonymity experiment on our physical ring signature for $\mathcal{A}$. Moreover, we have shown that $\Pr[b_* = 1 \text{ in } G_0] = \Pr[b_* = 1 \text{ in } G_3]$. Finally, we deduce that the probability that $\mathcal{A}$ returns 1 on the $(s, 0)$-anonymity experiment is equals to the probability that $\mathcal{A}$ returns 1 on the $(s, 1)$-anonymity experiment on our physical ring signature, which concludes the proof.

## 5 Conclusion

In this paper we have described a physical ring signature that is easy to set up and that uses everyday objects. We have built a prototype, and we believe that it can be used to explain in a playful way how a ring signature works to a public not familiar with cryptography. Some ring signatures have additional properties, such as linkability (any user can link two signatures produced by the same member) and traceability (an authority can lift anonymity in some cases). In future work, we would like to find ways to adapt our physical ring signature, or propose new ones, to achieve these properties.

## Acknowledgements

This work was supported by the ANR France 2030 "CyberINSA" (ANR-23-CMAS-0019).

## References

1. Masayuki Abe, Miyako Ohkubo, and Koutarou Suzuki. 1-out-of-n signatures from a variety of keys. In Yuliang Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*, pages 415–432. Springer, 2002.
2. Adam Bender, Jonathan Katz, and Ruggero Morselli. Ring signatures: Stronger definitions, and constructions without random oracles. *J. Cryptol.*, 22(1):114–138, 2009.
3. Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the weil pairing. *J. Cryptol.*, 17(4):297–319, 2004.
4. Xavier Bultel, Jannik Dreier, Jean-Guillaume Dumas, Pascal Lafourcade, Daiki Miyahara, Takaaki Mizuki, Atsuki Nagao, Tatsuya Sasaki, Kazumasa Shinagawa, and Hideaki Sone. Physical zero-knowledge proof for makaro. In Taisuke Izumi and Petr Kuznetsov, editors, *Stabilization, Safety, and Security of Distributed Systems - 20th International Symposium, SSS 2018, Tokyo, Japan, November 4-7, 2018, Proceedings*, volume 11201 of *Lecture Notes in Computer Science*, pages 111–125. Springer, 2018.
5. Ran Canetti and Ronald L. Rivest. Lecture 26. *Selected Topics in Cryptography*, 2004.
6. Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Yvo Desmedt, editor, *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, volume 839 of *Lecture Notes in Computer Science*, pages 174–187. Springer, 1994.
7. Jannik Dreier, Jean-Guillaume Dumas, Pascal Lafourcade, and Léo Robert. Optimal threshold padlock systems. *J. Comput. Secur.*, 30(5):655–688, 2022.
8. Jannik Dreier, Hugo Jonker, and Pascal Lafourcade. Secure auctions without cryptography. In Alfredo Ferro, Fabrizio Luccio, and Peter Widmayer, editors, *Fun with Algorithms - 7th International Conference, FUN 2014, Lipari Island, Sicily, Italy, July 1-3, 2014. Proceedings*, volume 8496 of *Lecture Notes in Computer Science*, pages 158–170. Springer, 2014.
9. Ronen Gradwohl, Moni Naor, Benny Pinkas, and Guy N. Rothblum. Cryptographic and physical zero-knowledge proof systems for solutions of sudoku puzzles. *Theory Comput. Syst.*, 44(2):245–268, 2009.
10. Yuichi Komano and Takaaki Mizuki. Coin-based secure computations. *Int. J. Inf. Sec.*, 21(4):833–846, 2022.
11. Pascal Lafourcade, Takaaki Mizuki, Atsuki Nagao, and Kazumasa Shinagawa. Light cryptography. In Lynette Drevin and Marianthi Theocharidou, editors, *Information Security Education. Education in Proactive Information Security - 12th IFIP WG 11.8 World Conference WISE 12, Lisbon, Portugal, June 25-27, 2019, Proceedings*, volume 557 of *IFIP Advances in Information and Communication Technology*, pages 89–101. Springer, 2019.
12. Takaaki Mizuki. Efficient and secure multiparty computations using a standard deck of playing cards. In Sara Foresti and Giuseppe Persiano, editors, *Cryptology and Network Security - 15th International Conference, CANS 2016, Milan, Italy, November 14-16, 2016, Proceedings*, volume 10052 of *Lecture Notes in Computer Science*, pages 484–499, 2016.
13. Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In Colin Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001, 7th International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, Australia, December 9-13, 2001, Proceedings*, volume 2248 of *Lecture Notes in Computer Science*, pages 552–565. Springer, 2001.

14. Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Paper 2004/332, 2004. `https://eprint.iacr.org/2004/332`.
15. Anh The Ta, Thanh Xuan Khuc, Tuong Ngoc Nguyen, Huy Quoc Le, Dung Hoang Duong, Willy Susilo, Kazuhide Fukushima, and Shinsaku Kiyomoto. Efficient unique ring signature for blockchain privacy protection. In Joonsang Baek and Sushmita Ruj, editors, *Information Security and Privacy - 26th Australasian Conference, ACISP 2021, Virtual Event, December 1-3, 2021, Proceedings*, volume 13083 of *Lecture Notes in Computer Science*, pages 391–407. Springer, 2021.
16. Sri Aravinda Krishnan Thyagarajan, Giulio Malavolta, Fritz Schmid, and Dominique Schröder. Verifiable timed linkable ring signatures for scalable payments for monero. In Vijayalakshmi Atluri, Roberto Di Pietro, Christian Damsgaard Jensen, and Weizhi Meng, editors, *Computer Security - ESORICS 2022 - 27th European Symposium on Research in Computer Security, Copenhagen, Denmark, September 26-30, 2022, Proceedings, Part II*, volume 13555 of *Lecture Notes in Computer Science*, pages 467–486. Springer, 2022.
17. Patrick P. Tsang and Victor K. Wei. Short linkable ring signatures for e-voting, e-cash and attestation. In Robert H. Deng, Feng Bao, HweeHwa Pang, and Jianying Zhou, editors, *Information Security Practice and Experience, First International Conference, ISPEC 2005, Singapore, April 11-14, 2005, Proceedings*, volume 3439 of *Lecture Notes in Computer Science*, pages 48–60. Springer, 2005.

## A Proof of Theorem 1

To prove Theorem 1, we prove the following two lemmas.

**Lemma 1.** *Let $n$ be an integer. For any $m \in \mathbb{Z}_n$ and any distinguisher $\mathcal{D}$, we have:*

$$\begin{aligned}&\Pr[x_1 \overset{\$}{\leftarrow} \mathbb{Z}_n; x_0 \leftarrow m - x_1 \bmod n; \ : \ 1 \leftarrow \mathcal{D}(x_0, x_1)] = \\ &\Pr[x_0 \overset{\$}{\leftarrow} \mathbb{Z}_n; x_1 \leftarrow m - x_0 \bmod n; \ : \ 1 \leftarrow \mathcal{D}(x_0, x_1)]\end{aligned}$$

*Proof.* In the first case, since $x_1 \overset{\$}{\leftarrow} \mathbb{Z}_n$ and $x_0 \leftarrow m - x_1 \bmod n$, each pair $(x_0, x_1) \in \mathbb{Z}_n$ such that $x_0 + x_1 = m \bmod n$ is generated with probability $1/n$. We remark that each of the $n$ pairs contains a different $x_0$, so each $x_0 \in \mathbb{Z}_n$ appears with probability $1/n$. Similarly, in the second case, if $x_0 \overset{\$}{\leftarrow} \mathbb{Z}_n$ and $x_1 \leftarrow m - x_0 \bmod n$, then each pair $(x_0, x_1) \in \mathbb{Z}_n$ such that $x_0 + x_1 = m \bmod n$ with a different $x_1$ is generated with probability $1/n$. We deduce that the two cases are indistinguishable, which concludes the proof.

**Lemma 2.** *Let $n$ and $s$ be two integers. For any $m \in \mathbb{Z}_n$, any pair $(i_0, i_1) \in \mathbb{Z}_s^2$ such that $i_0 \neq i_1$, and any distinguisher $\mathcal{D}$, we have:*

$$\begin{aligned}&\Pr\left[\begin{matrix}\forall i \in \mathbb{Z}_s \backslash \{i_0\}, x_i \overset{\$}{\leftarrow} \mathbb{Z}_n; \\ x_{i_0} \leftarrow \left(m - \sum_{i=0; i \neq i_0}^{s-1} x_i\right) \bmod n;\end{matrix} \ : \ 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s-1})\right] = \\ &\Pr\left[\begin{matrix}\forall i \in \mathbb{Z}_s \backslash \{i_1\}, x_i \overset{\$}{\leftarrow} \mathbb{Z}_n; \\ x_{i_1} \leftarrow \left(m - \sum_{i=0; i \neq i_1}^{s-1} x_i\right) \bmod n;\end{matrix} \ : \ 1 \leftarrow \mathcal{D}((x_i)_{i=0}^{s-1})\right]\end{aligned}$$

*Proof.* If $i_0 = i_1$, the result is trivial because the two expressions are the same. Else, by setting:

$$m' = \left(m - \sum_{i=0; i \notin \{i_0, i_1\}}^{s-1} x_i\right) \bmod n,$$

we have:

$$\begin{aligned} x_{i_b} = \left(m - \sum_{i=0; i \neq i_b}^{s-1} x_i\right) \bmod n &\Leftrightarrow x_{i_0} = m' - x_{i_1} \bmod n \quad (1) \\ &\Leftrightarrow x_{i_1} = m' - x_{i_0} \bmod n. \quad (2)\end{aligned}$$

We recall that in the two cases, each $x_i$ such that $i \notin \{i_0, i_1\}$ is generated at random. Therefore, these values cannot be used to distinguish between the two cases. If $x_{i_1}$ is chosen at random, then Equation 1 corresponds to the expression in the first probability in Lemma 1. Similarly, if $x_{i_0}$ is chosen at random, then Equation 2 corresponds to the expression in the second probability in Lemma 1. The values generated by these two expressions are therefore indistinguishable according to Lemma 1. Finally, the proof of Lemma 2 follows from Lemma 1.

Lemma 2 can easily be generalized to the case where the $x_i$ are vectors of integers, which leads directly to Theorem 1.