Monero's Building Blocks
Part 5 of 10 – *Cryptonote's linkable ring signature scheme*

Bassam El Khoury Seguias

BTC: 3FcVvBZwTUkUrcqJd16RcjR42qT2tDWHWn

ETH: 0xb79Fb9194C8Cc6221368bb70976e18609Ab9AcA8

March 13, 2018

1 Introduction

In this part, we introduce Monero's original signature scheme as described in van Saberhagen's seminal Cryptonote paper [2]. The scheme is an adaptation of the *Traceable Ring Signature* introduced by Fujisaki and Suzuki [1]. The most recent version of Monero implements a different signature known as RingCT. It modifies the original scheme to accomodate confidential transactions. We will discuss it in detail in parts 7, 8 and 9.

Security analysis of ring schemes consisted primarily in proving a) *correctness*, b) resilience against EFACM attacks in the RO model (i.e., *unforgeability*), and c) *anonymity* (i.e., signer ambiguity according to e.g., definition # 1 or # 2 as previously described in part 3). However, none of these security metrics tells if 2 signatures were generated by the same user or not. Doing so does not necessarily break the anonymity of the signer, but rather establishes a relationship between pairs of signatures. Identifying whether 2 signatures are linked or not is essential when dealing with electronic cash for example. In this case, the network must not tolerate the double spending of the same unit of electronic currency on 2 different transactions. In an electronic cash setting, the message typically consists of an unspent transaction output (also known as UTXO) and the objective is to make sure that the owner of a UTXO does not sign it twice (i.e., double spend it). Whenever this happens, the incident must be flagged and proper measures taken.

Monero in particular, and cryptocurrencies in general are prone to the double spending problem. This motivates the need to have an additional security requirement to tell if 2 signatures were issued by the same user. This must be done without releasing the identity of the user. We refer to the new requirement as *linkability*. It can commonly be achieved by adding to the ring signature a new signer-specific component known as a *tag* or a *key-image*.

Formally, we define a *linkable ring signature* scheme as a set of 4 algorithms:

1. The signer's key generation algorithm $\mathcal{G}$ (as described in part 1 of this series)
2. The ring signing algorithm Σ (as described in part 1 of this series).
3. The ring verification algorithm $\mathcal{V}$ (as described in part 1 of this series)
4. The ring linkability algorithm $\mathcal{L}$. Its input consists of a set of tags (key-images) and a given signature σ. It checks if σ's tag is included in the tag set. If so, it outputs *Linked*. Otherwise, it outputs *Independent* and adds the new tag to the set.

2 Cryptonote's original linkable ring signature

Cryptonote is an application layer protocol that supports a number of cryptocurrencies. The first implementation of the Cryptonote protocol dates back to July 2012 and consisted of a cryptocurrency known as Bytecoin (different than Bitcoin). In April 2014, Monero was launched as a fork of Bytecoin.

The Cryptonote scheme [2] relies on a large finite group generated by a particular elliptic curve whose equation is given by:

$$E : -x^2 + y^2 = 1 + dx^2y^2$$

For a concise introduction to elliptic curve cryptography, one could consult the post on *Elliptic Curve Groups*. The above equation is a polynomial over $\mathbb{F}_q$ where q is a very large prime and d is a pre-defined element of $\mathbb{F}_q$. To simplify the notation, we refer to the group generated by this elliptic curve as $E(\mathbb{F}_q)$. We note the following:

- Elements of $E(\mathbb{F}_q)$ are pairs $(x, y) \in \mathbb{F}_q^2$ that satisfy the above equation.
- Elliptic curve groups in general and $E(\mathbb{F}_q)$ in particular have a well defined addition operation that we denote by $\oplus$.
- $E(\mathbb{F}_q)$ contains a special element G (not necessarily unique) that we refer to as the base point. The base point has order $l < q$, where l is a very large prime. That means that adding G to itself l times yields the identity element e of $E(\mathbb{F}_q)$. In other terms, $G \oplus ... \oplus G = e$. We simply write $l \otimes G = e$ (the notation $\otimes$ serves as a reminder that this is scalar multiplication associated with $\oplus$).
- We let $\{G\}$ denote the group generated by G under the $\oplus$ operation of $E(\mathbb{F}_q)$. We also let $\{G\}^* \equiv \{G\} - e$.
- Solving the Discrete Logarithm (DL) problem on $\{G\}^*$ (and more generally on $E(\mathbb{F}_q)$) is thought to be intractable.

Cryptonote's signature uses 2 distinct hash functions $\mathcal{H}_1$ and $\mathcal{H}_2$ (modeled as 2 ROs). With a slight divergence from [2], we first introduce a hash function $\mathcal{H}_T$ before we define $\mathcal{H}_2$. The reason will become clearer in section 4 when we build the signing simulator to prove Cryptonote scheme's resilience against EFACM.

- $\mathcal{H}_1 : \{0,1\}^* \longrightarrow \mathbb{F}_q$
- $\mathcal{H}_T : \{G\}^* \longrightarrow \mathbb{F}_l^* \times \{G\}^*$

  $\mathcal{H}_T$ takes an element $s \in \{G\}^*$ and outputs a tuple $(v_s, v_s \otimes G) \in \mathbb{F}_l^* \times \{G\}^*$. Here v_s is a random element chosen according to a uniform distribution over $\mathbb{F}_l^*$. We then let $\mathcal{H}_2(s) \equiv v_s \otimes G$. So $\mathcal{H}_2 : \{G\}^* \longrightarrow \{G\}^*$, takes an element $s \in \{G\}^*$ and returns an element $v_s \otimes G \in \{G\}^*$ where v_s is randomly chosen in $\mathbb{F}_l^*$.

  Note that [2] defines $\mathcal{H}_2$ as a map from $E(\mathbb{F}_q)$ to $E(\mathbb{F}_q)$. Here we restricted the domain and the range to $\{G\}^*$ instead. This is because as we will see shortly, $\mathcal{H}_2$ is applied to public keys. Public keys are elements of $E(\mathbb{F}_q)$ that are scalar multiples of the base point G. Moreover, the scalar is never equal to order$(G) = l$ (we impose this constraint when we introduce the key generation algorithm $\mathcal{G}$ next). We are then justified in restricting the domain to $\{G\}^*$. The range is arbitrarily defined to be $\{G\}^*$, which is permissible since it preserves the injective nature of the map.

The scheme is defined by a set of 4 algorithms:

- **The key generation algorithm** $\mathcal{G}$. On input 1^k (k is the security parameter that by design we require to satisfy $k < log_2|\{G\}^*| = log_2(l-1)$), it produces a pair $(sk, pk) \equiv (x, y)$ of matching secret and public keys. x is randomly chosen in $\mathbb{F}_l^* \equiv \{1, ..., l-1\}$, and y is calculated as $x \otimes G$. (Note that G and y are both elements of $\{G\}^* \subset EC(\mathbb{F}_q)$ while x is an element of $\mathbb{F}_l^* \subset \mathbb{F}_q$.

  In addition to the (x, y) key pair, $\mathcal{G}$ computes $I \equiv x \otimes \mathcal{H}_2(y)$. I is known as the *key image* (or *tag*). It is signer-specific since it depends only on the signer's private and public keys. It allows the ring linkability algorithm $\mathcal{L}$ to test for independence between different signatures. $\mathcal{G}$ is modeled as a PPT Turing machine.

- **The ring signing algorithm** Σ. Suppose a user A_π decides to sign a message m on behalf of the ring of users $L \equiv \{A_1, ..., A_n\} \ni A_\pi$. A_π has a key pair given by (x_π, y_π) and a key-image (or tag) given by $I_\pi \equiv x_\pi \otimes \mathcal{H}_2(y_\pi)$. Σ does the following:

  1. $\forall i \in \{1, ...n\}, i \neq \pi$, choose random q_i, $w_i \in \{1, ...l\} \equiv \mathbb{F}_l$.
     Assign $L_i \equiv (q_i \otimes G) \oplus (w_i \otimes y_i)$ and $R_i \equiv (q_i \otimes \mathcal{H}_2(y_i)) \oplus (w_i \otimes I_\pi)$.
  2. Choose random $q_\pi \in \{1, ..., l\} \equiv \mathbb{F}_l$.
     Assign $L_\pi \equiv (q_\pi \otimes G)$ and $R_\pi \equiv (q_\pi \otimes \mathcal{H}_2(y_\pi))$.
  3. Assign $c \equiv \mathcal{H}_1(m, L_1, ..., L_n, R_1, ..., R_n)$.
  4. $\forall i \in \{1, ...n\}, i \neq \pi$, assign $c_i \equiv w_i$ and $r_i \equiv q_i$.
  5. Set $c_\pi \equiv c - \sum_{i \neq \pi} c_i \pmod{l}$ and $r_\pi \equiv q_\pi - c_\pi x_\pi \pmod{l}$. Here $c_\pi x_\pi$ denotes regular scalar multiplication in modulo l arithmetic.

  Σ outputs a signature $\sigma_\pi(m, L) \equiv (I_\pi, c_1, ..., c_n, r_1, ...r_n)$. Σ is a PPT algorithm.

- **The ring verification algorithm** $\mathcal{V}$. Given a ring signature σ, a message m, and the set $\{y_1, ...y_n\}$ of public keys of the ring members:

- *(Verification equations #1 to #2n)*: $\forall i \in \{1, ..., n\},\ \mathcal{V}$ computes
  $\{\ L'_i \equiv (r_i \otimes G)\ \oplus\ (c_i \otimes y_i)$
  $\{\ R'_i \equiv (r_i \otimes \mathcal{H}_2(y_i))\ \oplus\ (c_i \otimes I_\pi)$
- *(Verification equation #(2n + 1))*: $\mathcal{V}$ checks whether
  $\{\ \sum_i^n c_i = \mathcal{H}_1(m, L'_1, ...L'_n, R'_1, ...R'_n)\ (\text{mod } l)$

If equality holds, the signature is valid and $\mathcal{V}$ outputs *True*. Else, it outputs *False*.

- **The ring linkability algorithm** $\mathcal{L}$. It takes a $\mathcal{V}$-verified valid signature $\sigma_\pi(m, L)$. It checks if the key-image I_π was used in the past by comparing it to previous key-images stored in a set $\mathcal{I}$. If a match is found, then with overwhelming probability the 2 signatures were produced by the same key pair (as will be justified in the *exculpability* section), and $\mathcal{L}$ outputs *Linked*. Otherwise, its key-image is added to $\mathcal{I}$ and $\mathcal{L}$ outputs *Independent*.

3 Security analysis - Correctness

Let $\sigma_\pi(m, L) \equiv (I_\pi, c_1, ..., c_n, r_1, ..., r_n)$ be a Σ-generated signature. We compute:

1. $L'_i = (r_i \otimes G)\ \oplus\ (c_i \otimes y_i) =$
   - $(q_i \otimes G)\ \oplus\ (w_i \otimes y_i)$, if $i \neq \pi$ (since Σ dictates that $r_i \equiv q_i$ and $c_i \equiv w_i$).
   - $[(q_\pi - c_\pi x_\pi) \otimes G]\ \oplus\ (c_\pi \otimes y_\pi)$, if $i = \pi$ (since Σ dictates that $r_\pi \equiv q_\pi - c_\pi x_\pi$). And since $y_\pi = x_\pi \otimes G$, the resulting quantity becomes $q_\pi \otimes G$.

   We can then easily see that $\forall i \in \{1, ..., n\},\ L'_i = L_i$.

2. $R'_i = (r_i \otimes \mathcal{H}_2(y_i))\ \oplus\ (c_i \otimes I_\pi) =$
   - $(q_i \otimes \mathcal{H}_2(y_i))\ \oplus\ (w_i \otimes I_\pi)$, if $i \neq \pi$ (since Σ dictates that $r_i \equiv q_i$ and $c_i \equiv w_i$).
   - $[(q_\pi - c_\pi x_\pi) \otimes \mathcal{H}_2(y_\pi)]\ \oplus\ (c_\pi \otimes I_\pi)$, if $i = \pi$ (since Σ dictates that $r_\pi \equiv q_\pi - c_\pi x_\pi$). And since $I_\pi = x_\pi \otimes \mathcal{H}_2(y_\pi)$ (by $\mathcal{G}$'s construction), the resulting quantity becomes $q_\pi \otimes \mathcal{H}_2(y_\pi)$.

   We can then easily see that $\forall i \in \{1, ..., n\},\ R'_i = R_i$. Subsequently, we get:

   $$\mathcal{H}_1(m, L'_1, ..., L'_n, R'_1, ..., R'_n)\ (\text{mod } l) = \mathcal{H}_1(m, L_1, ..., L_n, R_1, ..., R_n)\ (\text{mod } l)$$

   $\equiv c \equiv \sum_{i=1}^n c_i$ (by construction of Σ). Hence Σ-generated signatures are valid.

4 Security analysis - Unforgeability vis-a-vis EFACM

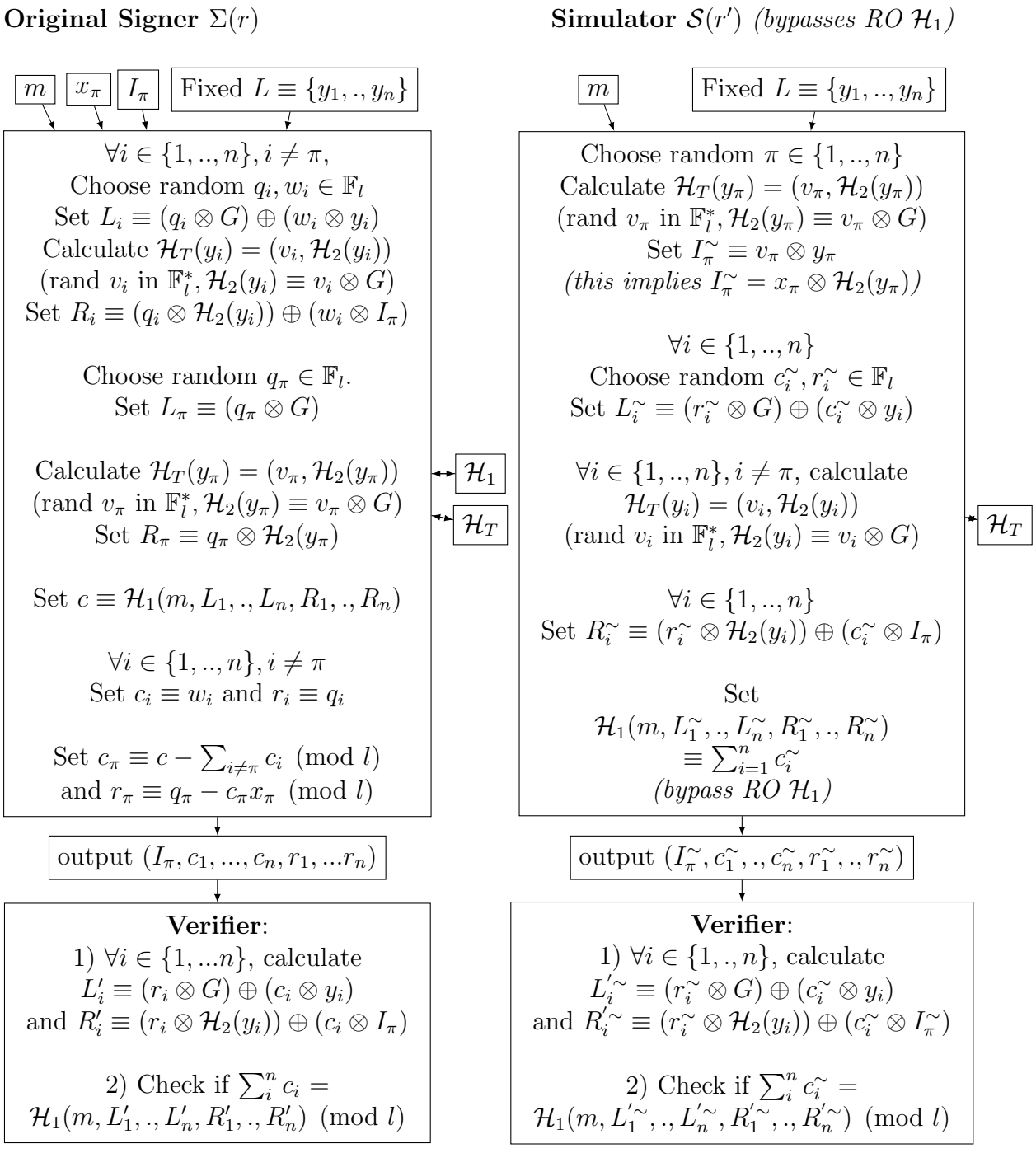
For unforgeability proofs, we follow the 5-step approach outlined earlier in part 1.

Step 1 : To prove that this scheme is secure against EFACM in the RO model, we proceed by contradiction and assume that there exists a PPT adversary $\mathcal{A}$ such that:

$$P_{\omega,r,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\Sigma^{\mathcal{H}_1,\mathcal{H}_T}(r)}\ succeeds\ in\ EFACM] = \epsilon(k), \text{ for } \epsilon \text{ non-negligible in } k.$$

Step 2 : Next, we build a simulator $\mathcal{S}(r')$ such that it:

- Does not have access to the private key of any signer.
- Has the same range as the original signing algorithm Σ (i.e., they output signatures taken from the same pool of potential signatures over all possible choices of RO functions and random tapes r' and r).
- Has indistinguishable probability distribution from that of Σ over this range.



The reason we introduced $\mathcal{H}_T$ as opposed to introducing only $\mathcal{H}_2$ is that the simulator makes use of the random element v_π in order to set $I_\pi^\sim$ to the desired value. In other words, the simulator needs to have access to the random element $v_\pi \in \mathbb{F}_l^*$ that is used in the calculation of $\mathcal{H}_2(y_\pi)$ in order to ensure that $I_\pi^\sim$ equates to $x_\pi \otimes \mathcal{H}_2(y_\pi)$.

By construction, the output of $\mathcal{S}$ will satisfy the verification equation. Moreover, it does its own random assignments to what otherwise would be calls to RO $\mathcal{H}_1$ (i.e., S bypasses RO $\mathcal{H}_1$). Next, note the following:

1. $\mathcal{S}$ does not use any private key.

2. Σ and $\mathcal{S}$ both have a range

$$R \equiv \{(\gamma, \epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n) \in \{G\}^* \times (\mathbb{F}_l)^{2n}$$

$$\text{s.t. } \textstyle\sum_{i=1}^{n} \epsilon_i = \mathcal{H}_1(m, L'_1, .., L'_n, R'_1, .., R'_n) \pmod{l},$$

$$\text{where } L'_i = (\beta_i \otimes G) \oplus (\epsilon_i \otimes y_i) \text{ and } R'_i = (\beta_i \otimes \mathcal{H}_2(y_i)) \oplus (\epsilon_i \otimes \gamma)\}.$$

3. Σ and $\mathcal{S}$ have the same probability distribution over R. Indeed, $\forall(\gamma, \epsilon_1, .., \epsilon_n, \beta_1, .., \beta_n) \in R$, we have:

   - For Σ :

$$P[(I_\pi, c_1, .., c_n, r_1, .., r_n) = (\gamma, \epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n)] =$$

$$P_{I_\pi \in \{G\}^*,\ c_i \in \mathbb{F}_l,\ r_i \in \mathbb{F}_l}[(I_\pi = \gamma) \cap (c_i = \epsilon_i,\ \forall i \in \{1, ..., n\}) \cap (r_i = \beta_i,\ \forall i \in \{1, ..., n\})]$$

$$= \tfrac{1}{|\{G\}^*|} \times (\tfrac{1}{l})^{2n} = \tfrac{1}{(l-1) \times l^{2n}}$$

   The first factor is the probability of choosing the exact I_π value in the set $\{G\}^*$ that is equal to γ. The second factor is the probability of choosing the exact $2n$ values given by the ϵ_i's and β_i's $\in \mathbb{F}_l$.

   - For $\mathcal{S}$:

$$P[(I_\pi^\sim, c_1^\sim, .., c_n^\sim, r_1^\sim, .., r_n^\sim) = (\gamma, \epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n)] =$$

$$P_{I_\pi^\sim \in \{G\}^*,\ c_i^\sim \in \mathbb{F}_l,\ r_i^\sim \in \mathbb{F}_l}[(I_\pi^\sim = \gamma) \cap (c_i^\sim = \epsilon_i,\ \forall i \in \{1, ..., n\}) \cap (r_i^\sim = \beta_i,\ \forall i \in \{1, ..., n\})]$$

$$= \tfrac{1}{|\{G\}^*|} \times (\tfrac{1}{l})^{2n} = \tfrac{1}{(l-1) \times l^{2n}}$$

   Note that the range of $I_\pi^\sim$ is equal to $\{G\}^*$ by construction of S. And so the first factor is the probability of choosing the exact $I_\pi^\sim$ value in the set $\{G\}^*$ that is equal to γ. The second factor is the probability of choosing the exact $2n$ values given by the ϵ_i's and β_i's $\in \mathbb{F}_l$.

With $\mathcal{S}$ adequately built, we conclude that (refer to section 6 of part 1 of this series for a justification):

$$P_{\omega,r,'\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\mathcal{S}^{\mathcal{H}_T}(r')} \textit{ succeeds in } EFACM] = \epsilon(k), \text{ for } \epsilon \text{ non-negligible in } k.$$

Step 3 : We now show that the probability of faulty collisions is negligible (refer to section 6 of part 1 of this series for an overview). The 2 tyes of collisions are:

- $Col_{Type\ 1}$: A tuple $(m, L_1, .., L_n, R_1, .., R_n)$ that $\mathcal{S}$ encounters – recall that $\mathcal{S}$ makes its own random assignment to $\mathcal{H}_1(m, L_1, .., L_n, R_1, .., R_n)$ and bypasses RO $\mathcal{H}_1$ – also appears in the list of queries that $\mathcal{A}(\omega)$ sends to RO $\mathcal{H}_1$. A conflict in the 2 values will happen with overwhelming probability and the execution will halt.
- $Col_{Type\ 2}$: A tuple $(m, L_1, .., L_n, R_1, .., R_n)$ that $\mathcal{S}$ encounters – recall that $\mathcal{S}$ makes its own random assignment to $\mathcal{H}_1(m, L_1, .., L_n, R_1, .., R_n)$ – is the same as another tuple $(m', L'_1, .., L'_n, R'_1, .., R'_n)$ that $\mathcal{S}$ encountered earlier – here too, $\mathcal{S}$ would have made its random assignment to $\mathcal{H}_1(m', L'_1, .., L'_n, R'_1, .., R'_n)$. Since the tuples are identical (i.e., $(m, L_1, .., L_n, R_1, .., R_n) = (m', L'_1, .., L'_n, R'_1, .., R'_n)$), the assignments must match (i.e., $\mathcal{H}_1(m, L_1, .., L_n, R_1, .., R_n) = \mathcal{H}_1(m', L'_1, .., L'_n, R'_1, .., R'_n)$). However, the likelihood that the 2 are equal is negligible. Hence they will be different with overwhelming probability and the execution will halt.

The aforementioned collisions must be avoided. In order to do so, we first calculate the probability of their occurence. We assume that during an EFACM attack, $\mathcal{A}(\omega)$ can make a maximum of Q_1 queries to RO $\mathcal{H}_1$, a maximum of Q_T queries to RO $\mathcal{H}_T$, and a maximum of Q_S queries to $\mathcal{S}(r')$. Q_1, Q_T, and Q_S are all assumed to be polynomial in the security parameter k, since the adversary is modeled as a PPT Turing machine.

$$P[Col_{Type\ 1}] = P[\cup_{all\ (m,L_1,..,R_n)}\{(m, L_1, .., R_n) \textit{ appeared in at least one of the } Q_S \textit{ queries to } \mathcal{S} \textit{ and } Q_1 \textit{ queries to RO } \mathcal{H}_1\}]$$

$$\leq P[\cup_{all\ L_1}\{L_1 \textit{ was part of at least one of the } Q_S \textit{ queries to } \mathcal{S} \textit{ and } Q_1 \textit{ queries to RO } \mathcal{H}_1\}]$$

$$\leq \sum_{all\ L_1 \in \{G\}} P[\cup_{(j=1,..,Q_S),\ (k=1,..,Q_1)}\{L_1 \textit{ was part of at least the } j^{th} \textit{ query to } \mathcal{S} \textit{ and } k^{th} \textit{ queries to RO } \mathcal{H}_1\}]$$

$$\leq \sum_{all\ L_1 \in \{G\}} \sum_{j=1}^{Q_S} \sum_{k=1}^{Q_1} P[L_1 \textit{ was part of at least the } j^{th} \textit{ query to } \mathcal{S} \textit{ and } k^{th} \textit{ queries to RO } \mathcal{H}_1]$$

$$\leq \sum_{all\ L_1 \in \{G\}} \sum_{j=1}^{Q_S} \sum_{k=1}^{Q_1} \frac{1}{|\{G\}|^2} = |\{G\}| \times \frac{Q_S Q_1}{|\{G\}|^2} = \frac{Q_S Q_1}{|\{G\}|} < \frac{Q_S Q_1}{2^k}.$$

$$(\textit{since } k < log_2(|\{G\}^*|) < log_2(|\{G\}|) \textit{ by design}).$$

Recalling that Q_S and Q_1 are polynomial in k, we conclude that $P[Col_{Type\ 1}]$ is negligible in k.

Next, we compute $P[Col_{Type\ 2}] =$

$$P[\cup_{all\ (m,L_1,..,R_n)}\{(m, L_1, .., R_n)\ appeared\ at\ least\ twice\ during\ queries\ to\ \mathcal{S}\}]$$

$$\leq P[\cup_{all\ L_1\in\{G\}}\{L_1\ was\ part\ of\ at\ least\ 2\ queries\ to\ \mathcal{S}\}]$$

$$\leq \sum_{L_1\in\{G\}} \binom{Q_S}{2} \times \frac{1}{|\{G\}|^2} < |\{G\}| \times \binom{Q_S}{2} \times \frac{1}{|\{G\}|^2} < \binom{Q_S}{2} \times \frac{1}{|\{G\}|} < \frac{Q_S^2}{2\times 2^k}.$$

$$(\text{since } k < log_2(|\{G\}^*|) < log_2(|\{G\}|) \text{ by design}).$$

Recalling that Q_S is polynomial in k, we conclude that $P[Col_{Type\ 2}]$ is negligible in k.

Putting it altogether, we find that the below quantity is negligible in k:

$$P[Col] = P[Col_{Type\ 1} \cup Col_{Type\ 2}] \leq \sum_{i=1}^{2} P[Col_{Type\ i}] \leq \frac{Q_S Q_1 + \frac{Q_S^2}{2}}{2^k} \equiv \delta(k)$$

This allows us to conclude that the below quantity is non-negligible in k (refer to section 6 of part 1 for a justification):

$$P_{\omega,r,'\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\mathcal{S}^{\mathcal{H}_T(r')}} succeeds\ in\ EFACM\ \cap \overline{Col}] \geq \epsilon(k) - \delta(k).$$

Step 4 : In this step, our objective is to show that if $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$ is a successful tuple that generated a first EFACM forgery, then the following quantity is non-negligible in k:

$$P_{\mathcal{H}_1}[\mathcal{A}(\omega^*)^{\mathcal{H}_1,\mathcal{H}_T^*,\mathcal{S}^{\mathcal{H}_T(r'^*)}}\ succeeds\ in\ EFACM\ \cap\ (\rho_{\alpha(\beta)} \neq \rho^*_{\alpha(\beta)})\ |$$
$$(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)\ is\ a\ succesfull\ first\ forgery,\ and\ (\rho_i = \rho_i^*)\ for\ i \in \{1, ...\alpha(\beta) - 1\}]$$

Here $\alpha(\beta)$ is an appropriate index that we will define in the proof. To further simplify the notation, we let $\rho_i^* \equiv \mathcal{H}_1^*(q_i^*)$ and $\rho_i \equiv \mathcal{H}_1(q_i)$ for all $i \in 1, .., \alpha(\beta)$. ($q_i$ and q_i^* denote respectively the i^{th} query to $\mathcal{H}_1$ and to $\mathcal{H}_1^*$).

Let's take a closer look at $P_{\omega,r,'\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\mathcal{S}^{\mathcal{H}_T(r')}} succeeds\ in\ EFACM \cap \overline{Col}]$.

Any successful forgery must pass the verification equation given by:

$$c \equiv \sum_{i=1}^{n} c_i = \mathcal{H}_1(m, L_1', ., L_n', R_1', ., R_n') \pmod{l}$$

$$\text{where } \forall i \in \{1, ...n\},\ L_i' \equiv (r_i \otimes G) \oplus (c_i \otimes y_i) \text{ and } R_i' \equiv (r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i \otimes I_\pi)$$

Notice that this equation takes the R_i''s as argument before verifying the equality. So we distinguish between 3 scenarios *(w.l.o.g. we assume that all $\mathcal{A}$-queries sent to RO $\mathcal{H}_1$ are distinct from each-other. Similarly, all $\mathcal{A}$-queries sent to RO $\mathcal{H}_T$ are distinct from each-other. This is because we can assume that $\mathcal{A}$ keeps a local copy of previous*

query results and avoid redundant calls):

- Scenario 1: $\mathcal{A}$ was successful in its forgery, and
  - No collisions occured, and
  - It never queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$.
- Scenario 2: $\mathcal{A}$ was successful in its forgery, and
  - No collisions occured, and
  - It queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ during execution, and
  - $\exists i \in \{1, .., n\}$ such that it queried RO $\mathcal{H}_T$ on input y_i after it had queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$.
- Scenario 3: $\mathcal{A}$ was successful in its forgery, and
  - No collisions occured, and
  - It queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ during execution, and
  - $\forall i \in \{1, .., n\}$, it queried RO $\mathcal{H}_T$ on input y_i before it queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$.

The probability of scenario 1 is upper-bounded by the probability that $\mathcal{A}$ picks its c_i's for $i \in \{1, .., n\}$ such that $c \equiv \sum_{i=1}^{n} c_i$ matches the value of $\mathcal{H}_1(m, L'_1, .., L'_n, R'_1, .., R'_n)$. Here, $\mathcal{H}_1(m, L'_1, .., L'_n, R'_1, .., R'_n)$ is the value that RO $\mathcal{H}_1$ returns to $\mathcal{V}$ (the verification algorithm) when verifying the validity of the forged signature. And since c can be any value in the range of $\mathcal{H}_1$ (which was defined to be $\mathbb{F}_q$) we get:

$$P[Scenario\ 1] \leq \frac{1}{q} < \frac{1}{l} = \frac{1}{|\{G\}|} < \frac{1}{|\{G\}^*|} \leq \frac{1}{2^k}, \text{ which is negligible in } k.$$

In scenario 2, let $i \in \{1, .., n\}$ be an index such that $\mathcal{A}$ queried RO $\mathcal{H}_T$ on input y_i after it had queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$. Note that during the verification process, $\mathcal{V}$ will calculate $R'_i \equiv (r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i \otimes I)$ and hence will make a call to $\mathcal{H}_T$ on input y_i (*remember that* $\mathcal{H}_2$ *is derived from* $\mathcal{H}_T$). The probability that the resulting R'_i matches the R'_i argument previously fed to $\mathcal{H}_1$ is upper-bounded by $\frac{1}{|\{G\}^*|}$ (since the range of $\mathcal{H}_2 = |\{G\}^*|$). Moreover, i can be any index in $\{1, .., n\}$. We get:

$$P[Scenario\ 2] \leq \frac{n}{|\{G\}^*|} \leq \frac{n}{2^k}, \text{ which is negligible in } k.$$

So we assume that a successful forgery will likely be of the Scenario 3 type.

$$\begin{gathered} P[Scenario\ 3] = \\ P_{\omega, r, {r'}_{\mathcal{H}_1, \mathcal{H}_T}}[\mathcal{A}(\omega)^{\mathcal{H}_1, \mathcal{H}_T, \mathcal{S}^{\mathcal{H}_T}(r')} succeeds\ in\ EFACM\ \cap \overline{Col}] - P[Scenario\ 1] \\ -P[Scenario\ 2] \end{gathered}$$

$$\geq \epsilon(k) - \delta(k) - \frac{1}{2^k} - \frac{n}{2^k} \equiv \nu(k), \text{ which is non-negligible in } k$$

Note that $\mathcal{A}(\omega)$ can send queries to RO $\mathcal{H}_1$ and RO $\mathcal{H}_T$ in any order it chooses to. This gives 2 different ways of referencing the index of a particular query sent to RO $\mathcal{H}_1$. One

way is to count the index as it appeared in the sequence of cumulative queries sent to both $\mathcal{H}_1$ and $\mathcal{H}_T$. In this case, indices take on values in $\{1, .., Q_1 + Q_T\}$. The other way, is to do the counting with respect to $\mathcal{H}_1$ queries only causing indices to take on values in $\{1, .., Q_1\}$. If i is the index counted in the cumulative numbering system (i.e., the former system), we let $\alpha(i)$ be the equivalent index in the latter system. Clearly, $\alpha(i) \leq i$.

We define $Ind(\omega, r', \mathcal{H}_1, \mathcal{H}_T)$ to be the index of the query $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ sent by $\mathcal{A}(\omega)$ to RO $\mathcal{H}_1$ during execution. Here, indexing is done with respect to the cumulative numbering system, and so $1 \leq Ind(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \leq (Q_1 + Q_T)$. We let $Ind(\omega, r', \mathcal{H}_1, \mathcal{H}_T) = \infty$ if query $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ was never asked by $\mathcal{A}(\omega)$. We also define the following condition:

$$E \equiv \{\forall i \in \{1, .., n\}, \mathcal{A}(\omega) \textit{ queried RO } \mathcal{H}_T \textit{ on input } y_i \textit{ before it queried RO } \mathcal{H}_1 \textit{ on input } (m, L'_1, .., L'_n, R'_1, .., R'_n)\}.$$

This definition allows us to build the following sets:

- $S=\{(\omega,r',\mathcal{H}_1,\mathcal{H}_T) \mid \mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\mathcal{S}^{\mathcal{H}_T(r')}} \textit{succeeds in EFACM} \cap \overline{Col} \cap E \cap Ind(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\neq\infty\}$

  In other terms, S is the set of tuples $(\omega, r', \mathcal{H}_1, \mathcal{H}_T)$ that yield a successful EFACM forgery when no collisions occur, and when $\mathcal{A}(\omega)$ queried RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ at some point during its execution such that condition E is met. This is none other than scenario 3 that was described earlier.

- $S_i=\{(\omega,r',\mathcal{H}_1,\mathcal{H}_T) \mid \mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\mathcal{S}^{\mathcal{H}_T(r')}} \textit{succeeds in EFACM} \cap \overline{Col} \cap E \cap Ind(\omega,r',\mathcal{H}_1,\mathcal{H}_T)=i\}$

  In other terms, S_i is the set of tuples $(\omega, r', \mathcal{H}_1, \mathcal{H}_T)$ that yield a successful EFACM forgery when no collisions occur, and when the index of the $\mathcal{A}(\omega)$-query sent to RO $\mathcal{H}_1$ on input $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ is equal to i, and such that condition E is met.

Recall that, $P_{\omega,r',\mathcal{H}_1,\mathcal{H}_T}[(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S] = P[Scenario\ 3] \geq \nu(k)$, (non-negligible in k).

Clearly, $\{\cup_{i=1}^{Q_1+Q_T} S_i\}$ partitions S. So $\sum_{i=1}^{Q_1+Q_T} P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] = 1$.

This implies that $\exists i \in \{1,..,Q_1+Q_T\}$ $s.t.$ $P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] \geq \frac{1}{2(Q_1+Q_T)}$.

If this were not the case, then one would get the following contradiction:

$$1 = \sum_{i=1}^{Q_1+Q_T} P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] < (Q_1+Q_T)\times\frac{1}{2(Q_1+Q_T)} = \frac{1}{2} < 1.$$

So we introduce the set I consisting of all indices that meet the $\frac{1}{2(Q_1+Q_T)}$ threshold, i.e.

$$I = \{i \in \{1, .., Q_1 + Q_T\}\ s.t\ P[(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S_i \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S] \geq \frac{1}{2(Q_1+Q_T)}\}$$

We claim that $P[Ind(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in I \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S] \geq \frac{1}{2}$.

Proof: By definition of the sets S_i, we have:

$$P[Ind(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in I \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] = \sum_{i\in I} P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S]$$

$$= 1-\sum_{j\notin I} P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_j \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] > 1-\sum_{j\notin I} \frac{1}{2(Q_1+Q_T)} > 1-\frac{Q_1+Q_T}{2(Q_1+Q_T)} = \frac{1}{2}$$

The next step is to apply the splitting lemma to each $S_i,\ i \in I$. First note that:

$$P_{\omega,r',\mathcal{H}_1,\mathcal{H}_T}[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i] = P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S] \times P_{\omega,r',\mathcal{H}_1,\mathcal{H}_T}[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S]$$

$$\geq \frac{1}{2(Q_1+Q_T)} \times \nu(k)$$

Referring to the notation used in the splitting lemma (section 7 of part 1), we let:

$\{\ A \equiv S_i$

$\{\ X \equiv (\omega, r', \rho_1, ..., \rho_{\alpha(i)-1}, \mathcal{H}_T)$

$\{\ Y \equiv (\rho_{\alpha(i)}, ..., \rho_{Q_1})$

$\{\ \epsilon \equiv \frac{\nu(k)}{2(Q_1+Q_T)}$

$\{\ \alpha \equiv \frac{\nu(k)}{4(Q_1+Q_T)} = \frac{\epsilon}{2}$

X is defined as the space of tuples of:

- All random tapes ω
- All random tapes r'
- All possibe RO $\mathcal{H}_1$ answers to the first $(\alpha(i)-1)$ queries sent by $\mathcal{A}(\omega)$ *(note the usage of α-indexing since indexing is done with respect to RO $\mathcal{H}_1$ queries only)*
- All RO $\mathcal{H}_T$ *(this means all possible RO $\mathcal{H}_T$ answers to the Q_T queries sent by $\mathcal{A}(\omega)$).*

Y is defined as the space of all possible RO $\mathcal{H}_1$ answers to the last $(Q_1 - \alpha(i) + 1)$ queries sent by $\mathcal{A}(\omega)$. (Recall that $\rho_j \equiv \mathcal{H}_1(q_j)$ where q_j is the j^{th} query sent to RO $\mathcal{H}_1$).

The splitting lemma guarantees the existence of a subset Ω_i of tuples $(\omega, r', \mathcal{H}_1, \mathcal{H}_T)$ such that:

- $P_{\omega,r',\mathcal{H}_1,\mathcal{H}_T}[(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in \Omega_i] \geq \frac{\nu(k)}{4(Q_1+Q_T)}$
- $\forall[(\omega^\sim, r'^\sim, \mathcal{H}_1^\sim, \mathcal{H}_T^\sim) \equiv (\omega^\sim, r'^\sim, \rho_1^\sim, ., \rho_{\alpha(i)-1}^\sim,\ \rho_{\alpha(i)}^\sim, ., \rho_{Q_1}^\sim, \mathcal{H}_T^\sim)] \in \Omega_i$, we have

$$P_{\mathcal{H}_1}[(\omega^\sim,r'^\sim,\ \rho_1^\sim,.,\rho_{\alpha(i)-1}^\sim,\ \rho_{\alpha(i)},.,\rho_{Q_1},\mathcal{H}_T^\sim)\in S_i \mid (\omega^\sim,r'^\sim,\mathcal{H}_1^\sim,\mathcal{H}_T^\sim)\in\Omega_i] \geq \frac{\nu(k)}{4(Q_1+Q_T)}, \text{ and so}$$

$$P_{\mathcal{H}_1}[(\omega^\sim,r'^\sim,\mathcal{H}_1,\mathcal{H}_T^\sim)\in S_i \mid (\omega^\sim,r'^\sim,\mathcal{H}_1^\sim,\mathcal{H}_T^\sim)\in\Omega_i,\ \rho_1=\rho_1^\sim\ ,...,\ \rho_{\alpha(i)-1}=\rho_{\alpha(i)-1}^\sim)] \geq \frac{\nu(k)}{4(Q_1+Q_T)}$$

- $P[(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in \Omega_i \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S_i] \ \geq\ (\frac{\nu(k)}{4(Q_1,+Q_T)})/(\frac{\nu(k)}{2(Q_1+Q_T)}) \ =\ \frac{1}{2}$

We would like to compute the probability of finding a 2^{nd} successful tuple $(\omega^*, r'^*, \mathcal{H}_1^{\sim}, \mathcal{H}_T^*)$ given that $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$ was a successful 1^{st} tuple and such that $\rho_j^{\sim} = \rho_j^*,\ \forall j \in \{1, .., \alpha(i) - 1\}$. That means finding the following probability:

$$P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_i \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in S_i,\ \rho_1 = \rho_1^*, ..., \rho_{\alpha(i)-1} = \rho_{\alpha(i)-1}^*].$$

From the splitting lemma results, we have a (non-negligible in k) lower-bound on $P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_i \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in \Omega_i,\ \rho_1 = \rho_1^*, ..., \rho_{\alpha(i)-1} = \rho_{\alpha(i)-1}^*]$.

Note however, that Ω_i and S_i are generally distinct sets. And so we **cannot** conclude that

$$P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_i \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in S_i,\ \rho_1 = \rho_1^*, ..., \rho_{\alpha(i)-1} = \rho_{\alpha(i)-1}^*]$$

$$= P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_i \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in \Omega_i,\ \rho_1 = \rho_1^*, ..., \rho_{\alpha(i)-1} = \rho_{\alpha(i)-1}^*]$$

and therefore we **cannot** conclude that the following quantity is non-negligible in k

$$P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_i \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in S_i,\ \rho_1 = \rho_1^*, ..., \rho_{\alpha(i)-1} = \rho_{\alpha(i)-1}^*]$$

In order to show that the above quantity is non-negligible in k, we proceed differently. Suppose we can show that the following probability is non-negligible in k:

$$P_{(\omega, r', \mathcal{H}_1, \mathcal{H}_T)}[\exists \beta \in I \ s.t.\ (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_\beta \cap S_\beta)]$$

This would imply that with non-negligible probability, we can find a tuple that belongs to S_β (and hence corresponds to a successful forgery) and at the same time belongs to Ω_β. We can then invoke the splitting lemma result just mentioned, to find a second tuple coresponding to a second forgery and that has the desired properties.

To prove the above, we proceed as follows:

$$P[\exists \beta \in I \ s.t.\ (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_\beta \cap S_\beta) \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S]$$

$$= P[\cup_{i \in I} \{(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_i \cap S_i) \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S\}]$$

$= \sum_{i \in I} P[(\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_i \cap S_i) \mid (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in S]$, since the S_i's are disjoint.

$$= \sum_{i \in I} \{\ P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in\Omega_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in(S_i\cap S)] \times P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S]\ \}$$

$$= \sum_{i \in I} \{\ P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in\Omega_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i] \times P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S]\ \}$$

$$\geq \tfrac{1}{2} \sum_{i \in I} P[(\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S_i \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T)\in S],\ \textit{(3^{rd} result of splitting lemma above)}$$

$$\geq \tfrac{1}{2} \times \tfrac{1}{2}\ \textit{(by the claim proven earlier)} = \tfrac{1}{4}.$$

And so we conclude that:

$$P_{(\omega,r',\mathcal{H}_1,\mathcal{H}_T)}[\exists \beta \in I \ s.t. \ (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_\beta \cap S_\beta)]$$

$$P_{(\omega,r',\mathcal{H}_1,\mathcal{H}_T)}[\exists \beta \in I \ s.t. \ (\omega, r', \mathcal{H}_1, \mathcal{H}_T) \in (\Omega_\beta \cap S_\beta \cap S)]$$

$$= P[\exists \beta \in I \ s.t. \ (\omega,r',\mathcal{H}_1,\mathcal{H}_T) \in (\Omega_\beta \cap S_\beta) \mid (\omega,r',\mathcal{H}_1,\mathcal{H}_T) \in S] \times P_{(\omega,r',\mathcal{H}_1,\mathcal{H}_T)}[(\omega,r',\mathcal{H}_1,\mathcal{H}_T) \in S] \geq \frac{\nu(k)}{4}$$

which is non-negligible in k.

So let β be such an index and $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$ such a tuple. From the result above, we know that finding such a $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in (\Omega_\beta \cap S_\beta)$ can be done with non-negligible probability. And since $(\Omega_\beta \cap S_\beta) \subset \Omega_\beta$, we must have $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in \Omega_\beta$. We can then invoke the 2^{nd} consequence of the splitting lemma and write:

$$P_{\mathcal{H}_1}[(\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_\beta \mid (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in S_\beta, \ \rho_1 = \rho_1^*, .., \rho_{\alpha(\beta)-1} = \rho_{\alpha(\beta)-1}^*)] =$$

$$P_{\mathcal{H}_1}[(\omega^*,r'^*,\mathcal{H}_1,\mathcal{H}_T^*) \in S_\beta \mid (\omega^*,r'^*,\mathcal{H}_1^*,\mathcal{H}_T^*) \in \Omega_\beta, \ \rho_1=\rho_1^*,..,\rho_{\alpha(\beta)-1} = \rho_{\alpha(\beta)-1}^*)] \geq \frac{\nu(k)}{4(Q_1+Q_T)}$$

We still have one last constraint to impose and that is that $\rho_{\alpha(\beta)}^* \neq \rho_{\alpha(\beta)}$. We show that the following quantity is non-negligible:

$$P_{\mathcal{H}_1}[((\omega^*, r'^*, \mathcal{H}_1, \mathcal{H}_T^*) \in S_\beta) \ \cap \ (\rho_{\alpha(\beta)} \neq \rho_{\alpha(\beta)}^*) | \ (\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in S_\beta, \ \rho_1 = \rho_1^*, ..., \rho_{\alpha(\beta)-1} = \rho_{\alpha(\beta)-1}^*)]$$

To prove this, we use the same technique employed in parts 2 and 4 of this series. Note that if B and C are independent events, then we can write:

$$P[A|C] = P[A \cap B|C] + P[A \cap \overline{B}|C] \leq P[A \cap B|C] + P[\overline{B}|C] = P[A \cap B|C] + P[\overline{B}]$$

And so we get $P[A \cap B|C] \geq P[A|C] - P[\overline{B}]$.

This result allows us to write:

$$P_{\mathcal{H}_1}[((\omega^*,r'^*,\mathcal{H}_1,\mathcal{H}_T^*) \in S_\beta) \cap (\rho_{\alpha(\beta)} \neq \rho_{\alpha(\beta)}^*) \mid (\omega^*,r'^*,\mathcal{H}_1^*,\mathcal{H}_T^*) \in S_\beta, \ \rho_1=\rho_1^* ..., \ \rho_{\alpha(\beta)-1}=\rho_{\alpha(\beta)-1}^*)]$$

$$\geq P_{\mathcal{H}_1}[(\omega^*,r'^*,\mathcal{H}_1,\mathcal{H}_T^*) \in S_\beta \mid (\omega^*,r'^*,\mathcal{H}_1^*,\mathcal{H}_T^*) \in S_\beta, \ \rho_1=\rho_1^* \ ,..., \ \rho_{\alpha(\beta)-1}=\rho_{\alpha(\beta)-1}^*)] - P_{\mathcal{H}_1}[\rho_{\alpha(\beta)} = \rho_{\alpha(\beta)}^*]$$

$$= P_{\mathcal{H}_1}[(\omega^*,r'^*,\mathcal{H}_1,\mathcal{H}_T^*) \in S_\beta \mid (\omega^*,r'^*,\mathcal{H}_1^*,\mathcal{H}_T^*) \in \Omega_\beta, \ \rho_1=\rho_1^* \ ,..., \ \rho_{\alpha(\beta)-1}=\rho_{\alpha(\beta)-1}^*)] - P_{\mathcal{H}_1}[\rho_{\alpha(\beta)} = \rho_{\alpha(\beta)}^*]$$

(because we chose $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in \Omega_\beta \cap S_\beta$*)*

$$\geq \frac{\nu(k)}{4(Q_1+Q_T)} - \frac{1}{2^k}, \text{ which is non-negligible in } k.$$

Step 5 : The final step uses the 2 forgeries obtained earlier to solve an instance of the Discrete Logarithm (DL) problem. Here is a recap of Step 4 results:

- With non-negligible probability of at least $\frac{\nu(k)}{4}$ we get a successful tuple $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$, s.t. $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in (\Omega_\beta \cap S_\beta)$ for some index $\beta \in I$. By

running $\mathcal{A}$ a number of times polynomial in k, we can find such a tuple.

- Once we find such a tuple, we've also shown that with non-negligible probability of at least $\frac{\nu(k)}{4(Q_1+Q_T)} - \frac{1}{2^k}$, we can find another successful tuple $(\omega^*, r'^*, \mathcal{H}_1^\sim, \mathcal{H}_T^*)$ such that $(\omega^*, r'^*, \mathcal{H}_1^\sim, \mathcal{H}_T^*) \in S_\beta$ and $(\rho_1^\sim = \rho_1^*), .., (\rho_{\alpha(\beta)-1}^\sim = \rho_{\alpha(\beta)-1}^*),\ (\rho_{\alpha(\beta)}^\sim \neq \rho_{\alpha(\beta)}^*)$.

Let $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$ correspond to forgery $\sigma_a(m_a, L) \equiv (I_a, (c_1)_a, .., (c_n)_a, (r_1)_a, .., (r_n)_a)$, and $(\omega^*, r'^*, \mathcal{H}^\sim, \mathcal{H}_T^*)$ correspond to forgery $\sigma_b(m_b, L) \equiv (I_b, (c_1)_b, .., (c_n)_b, (r_1)_b, .., (r_n)_b)$.

Recall that $\alpha(\beta)$ is the index of the query $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ that $\mathcal{A}$ sends to RO $\mathcal{H}_1$. Since the 2 experiments corresponding to the 2 successful tuples have:

- The same random tapes ω^* and r'^*
- The same RO $\mathcal{H}_T^*$
- ROs $\mathcal{H}_1^*$ and $\mathcal{H}_1^\sim$ behave the same way on the first $\alpha(\beta) - 1$ queries,

we can be confident that the first $\alpha(\beta)$ queries sent to the 2 ROs $\mathcal{H}_1^*$ and $\mathcal{H}_1^\sim$ are identical.

In particular the two $[\alpha(\beta)]^{th}$ queries are the same. And so:

$$(m_a, (L'_1)_a, .., (L'_n)_a, (R'_1)_a, .., (R'_n)_a) = (m_b, (L'_1)_b, .., (L'_n)_b, (R'_1)_b, .., (R'_n)_b)$$

$$\implies \forall i \in \{1, .., n\},\ (L'_i)_a = (L'_i)_b$$

$$\implies \forall i \in \{1, .., n\},\ ((r_i)_a \otimes G) \oplus ((c_i)_a \otimes y_i) = ((r_i)_b \otimes G) \oplus ((c_i)_b \otimes y_i),$$

$$\implies \forall i \in \{1, .., n\},\ x_i[(c_i)_a - (c_i)_b] = (r_i)_b - (r_i)_a \pmod{l} \text{ (by writing } y_i = x_i \otimes G)$$

Moreover, we have

$$\sum_{i=1}^n (c_i)_a = \mathcal{H}_1^*(m_a, (L'_1)_a, ., (L'_n)_a, (R'_1)_a, ., (R'_n)_a) \pmod{l} \textit{ (since } \sigma_a \textit{ is a valid forgery)}$$

$$= \rho_{\alpha(\beta)}^* \neq \rho_{\alpha(\beta)}^\sim \textit{ (by design of the forgery tuples)}$$

$$= \mathcal{H}_1^\sim(m_b, (L'_1)_b, ., (L'_n)_b, (R'_1)_b, ., (R'_n)_b) \pmod{l} = \sum_{i=1}^n (c_i)_b \textit{ (since } \sigma_b \textit{ is a valid forgery)}$$

Since $\sum_{i=1}^n (c_i)_a \neq \sum_{i=1}^n (c_i)_b$, we conclude that $\exists j \in \{1, .., n\}$ *s.t.* $(c_j)_a \neq (c_j)_b$

That means that we can solve for $x_j = \frac{(r_j)_b - (r_j)_a}{(c_j)_a - (c_j)_b} \pmod{l}$ in polynomial time, contradicting the intractability of DL on elliptic curve groups. We conclude that the signature scheme is secure against EFACM in the RO model.

5 Security analysis - Exculpability

We encountered the notion of *exculpability* when we introduced the 2 *anonymity* definitions in part 3 of this series. In that context, we said that a signer is exculpable if

her identity can not be established even if her private key gets compromised. In other terms, no one can prove that she was the actual signer under any circumstance. This ensures her exculpability. In this section, we introduce a different notion of exculpability described in [1]. It has to do with *unforgeability* as opposed to *anonymity*.

Exculpability Suppose $(n-1)$ private keys have been compromised in an n-ring setting. Let π denote the index of the only non-compromised private key x_π, and let I_π denote the key-image (or tag) associated with the key pair (x_π, y_π). We investigate whether it is likely to produce a valid forgery with key-image I_π. In what follows, we show that this can only happen with negligible probability. In essence, this means that a non-compromised **honest** ring member (*by honest we mean a ring member that signs at most once using his private key*) does not run the risk of encountering a forged signature that carries his key-image. In the context of Cryptonote, this implies that a non-compromised **honest** ring member cannot be accused of signing twice using the same key image or tag, and hence is exculpable.

Note that since the adversary $\mathcal{A}(\omega)$ has access to the $(n-1)$ compromised private keys, it can easily calculate their corresponding public keys. Doing so will allow it to identify the public key y_π of the non-compromised ring member. That means that it can determine the index π of the non-compromised member in the ring $L \equiv \{y_1, .., y_n\}$. In order to prove the exculpability of the Cryptonote scheme, we follow an almost identical proof to that of the previous section (i.e., unforgeability vis-a-vis EFACM) and apply the same 5-step approach.

Step 1 : We proceed by contradiction and assume that there exists a PPT adversary $\mathcal{A}$ such that:

$$P_{\omega,r,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\{x_1,..,\hat{x_\pi},..,x_n\},\Sigma^{\mathcal{H}_1,\mathcal{H}_T}(r)} \textit{ succeeds in creating a forgery } \sigma(m,L) \equiv (I_\pi, c_1, .., c_n, r_1, ..r_n)] = \epsilon(k), \text{ for } \epsilon \text{ non-negligible in } k.$$

We refer to the event *succeeds in creating a forgery* $\sigma(m,L) \equiv (I_\pi, c_1, .., c_n, r_1, ..r_n)$ as *succeeds in* $EFACM_{Ex_\pi}$. We re-write the above equation as:

$$P_{\omega,r,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\{x_1,..,\hat{x_\pi},..,x_n\},\Sigma^{\mathcal{H}_1,\mathcal{H}_T}(r)} \textit{ succeeds in } EFACM_{Ex_\pi}] = \epsilon(k), \text{ for } \epsilon \text{ non-negligible in } k.$$

The notation used makes it explicit that $\mathcal{A}(\omega)$ can access the set of compromised keys $\{x_1, .., \hat{x_\pi}, .., x_n\}$ with x_π excluded. Success is defined as issuing a forged signature with key image or tag equal to $I_\pi \equiv x_\pi \otimes \mathcal{H}_2(y_\pi)$. (*Recall that* $\mathcal{H}_2$ *is derived from* $\mathcal{H}_T$).

Step 2 : The next step consists in building a simulator $\mathcal{S}(r')$ such that it:

- Does not have access to the private key of any signer.
- Has the same range as the original signing algorithm Σ (i.e., they output signatures taken from the same pool of potential signatures over all possible choices of RO functions and respective random tapes r' and r).

- Has indistinguishable probability distribution from that of Σ over this range.

The simulator $S(r')$ is the same as the one we built in the previous section. The only nuance is that $S(r')$ does not choose a random index π, since $\mathcal{A}(\omega)$ already knows the index of the non-compromised ring member.

Step 3 : The logical reasoning and procedure are identical to those of the previous section. We conclude that

$$P_{\omega,r,'\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\{x_1,..,\hat{x_\pi},..,x_n\},\mathcal{S}^{\mathcal{H}_T(r')}} succeeds\ in\ EFACM_{Ex_\pi}\ \cap \overline{Col}] \geq \epsilon(k) - \delta(k).$$

Step 4 : Here too, the logical reasoning and procedure are identical to those of the previous section. In particular, we define the following sets in a similar way:

- $S=$
  $\{(\omega,r',\mathcal{H}_1,\mathcal{H}_T)|\ \mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\{x_1,..,\hat{x_\pi},..,x_n\}},\ \mathcal{S}^{\mathcal{H}_T(r')} succeeds\ in\ EFACM_{Ex_{x_\pi}} \cap \overline{Col} \cap E \cap\ Ind(\omega,r',\mathcal{H}_1,\mathcal{H}_T) \neq \infty\}$

- $S_i=$
  $\{(\omega,r',\mathcal{H}_1,\mathcal{H}_T)|\ \mathcal{A}(\omega)^{\mathcal{H}_1,\mathcal{H}_T,\{x_1,..,\hat{x_\pi},..,x_n\}},\ \mathcal{S}^{\mathcal{H}_T(r')} succeeds\ in\ EFACM_{Ex_{x_\pi}} \cap \overline{Col} \cap E \cap\ Ind(\omega,r',\mathcal{H}_1,\mathcal{H}_T) = i\}$

and conclude that:

$$P_{\mathcal{H}_1}[((\omega^*,r'^*,\mathcal{H}_1,\mathcal{H}_T^*) \in S_\beta) \cap (\rho_{\alpha(\beta)} \neq \rho^*_{\alpha(\beta)}) \mid (\omega^*,r'^*,\mathcal{H}_1^*,\mathcal{H}_T^*) \in S_\beta,\ \rho_1 = \rho_1^* ...,\ \rho_{\alpha(\beta)-1} = \rho^*_{\alpha(\beta)-1})]$$

$$\geq \frac{\nu(k)}{4(Q_1+Q_T)} - \frac{1}{2^k}, \text{ which is non-negligible in } k.$$

Here $\alpha(\beta)$, as before, is an appropriately defined index, $\rho_i^* \equiv \mathcal{H}_1^*(q_i)$, and $\rho_i \equiv \mathcal{H}_1(q_i)$ for all $i \in 1,..,\alpha(\beta)$.(q_i denotes the i^{th} query sent to RO).

Step 5 : The final step uses the 2 forgeries obtained earlier to solve an instance of the Discrete Logarithm (DL) problem. Here is a recap of Step 4 results:

- With non-negligible probability of at least $\frac{\nu(k)}{4}$ we get a successful tuple $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$, s.t. $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*) \in (\Omega_\beta \cap S_\beta)$ for some index $\beta \in I$. By running $\mathcal{A}$ a number of times polynomial in k, we can find such a tuple.

- Once we find such a tuple, we've also shown that with non-negligible probability of at least $\frac{\nu(k)}{4(Q_1+Q_T)} - \frac{1}{2^k}$, we can find another successful tuple $(\omega^*, r'^*, \mathcal{H}_1^\sim, \mathcal{H}_T^*)$ such that $(\omega^*, r'^*, \mathcal{H}_1^\sim, \mathcal{H}_T^*) \in S_\beta$ and $(\rho_1^\sim = \rho_1^*), ..., (\rho^\sim_{\alpha(\beta)-1} = \rho^*_{\alpha(\beta)-1}),\ (\rho^\sim_{\alpha(\beta)} \neq \rho^*_{\alpha(\beta)})$.

Let $(\omega^*, r'^*, \mathcal{H}_1^*, \mathcal{H}_T^*)$ correspond to forgery $\sigma_a(m_a, L) \equiv (I_\pi, (c_1)_a, .., (c_n)_a, (r_1)_a, .., (r_n)_a)$, and $(\omega^*, r'^*, \mathcal{H}^\sim, \mathcal{H}_T^*)$ correspond to forgery $\sigma_b(m_b, L) \equiv (I_\pi, (c_1)_b, .., (c_n)_b, (r_1)_b, .., (r_n)_b)$.

Recall that $\alpha(\beta)$ is the index of the query $(m, L'_1, .., L'_n, R'_1, .., R'_n)$ that $\mathcal{A}$ sends to RO $\mathcal{H}_1$. Since the 2 experiments corresponding to the 2 successful tuples have:

- The same random tapes ω^* and r'^*
- The same RO $\mathcal{H}_T^*$
- ROs $\mathcal{H}_1^*$ and $\mathcal{H}_1^\sim$ behave the same way on the first $\alpha(\beta) - 1$ queries,

we can be confident that the first $\alpha(\beta)$ queries sent to the 2 ROs $\mathcal{H}_1^*$ and $\mathcal{H}_1^\sim$ are identical.

In particular the two $[\alpha(\beta)]^{th}$ queries are the same. And so:

$$(m_a, (L'_1)_a, .., (L'_n)_a, (R'_1)_a, .., (R'_n)_a) = (m_b, (L'_1)_b, .., (L'_n)_b, (R'_1)_b, .., (R'_n)_b)$$

$$\implies \forall i \in \{1, .., n\},\ (L'_i)_a = (L'_i)_b, \text{ and } (R'_i)_a = (R'_i)_b$$

Let $R'_i \equiv (R'_i)_a = (R'_i)_b$, and $L'_i \equiv (L'_i)_a = (L'_i)_b$. For each $i \in \{1, .., n\}$, we get 2 identical systems of 2 equations dictated by $\mathcal{V}$'s verification computation:

First system of 2 linear equations **Second system of 2 linear equations**

$$\{\ R'_i = ((r_i)_a \otimes \mathcal{H}_2(y_i)) \oplus ((c_i)_a \otimes I_\pi) \qquad \{\ R'_i = ((r_i)_b \otimes \mathcal{H}_2(y_i)) \oplus ((c_i)_b \otimes I_\pi)$$

$$\{\ L'_i = ((r_i)_a \otimes G) \oplus ((c_i)_a \otimes y_i) \qquad \{\ L'_i = ((r_i)_b \otimes G) \oplus ((c_i)_b \otimes y_i)$$

$\forall i \in \{1, .., n\}$, the first system is a linear system of 2 equations in variables $(r_i)_a$ and $(c_i)_a$. Similarly, the second system is a linear system of 2 equations in variables $(r_i)_b$ and $(c_i)_b$. The 2 systems are identical with different variable names. Hence, if $((r_i^*)_a, (c_i^*)_a)$ is a unique solution to the first system and $((r_i^*)_b, (c_i^*)_b)$ a unique solution to the second, we can be confident that $(r_i^*)_a = (r_i^*)_b$ and $(c_i^*)_a = (c_i^*)_b$. (Note that when we previously proved resilience against EFACM in section 4, the 2 forged signatures did not necessarily share the same tag I_π and so the 2 systems of linear equations would have been different from each other). For either system to admit a unique solution, the 2 equations must be linearly independent. We re-write the 2 systems as follows:

First system of 2 linear equations **Second system of 2 linear equations**

$$\{\ R'_i = ((r_i)_a \otimes \mathcal{H}_2(y_i)) \oplus ((c_i)_a \otimes I_\pi) \qquad \{\ R'_i = ((r_i)_b \otimes \mathcal{H}_2(y_i)) \oplus ((c_i)_b \otimes I_\pi)$$

$$\{\ log_G(L'_i) = (r_i)_a\ +\ (c_i)_a x_i \qquad \{\ log_G(L'_i) = (r_i)_b\ +\ (c_i)_b x_i$$

If we multiply the second equation by $\mathcal{H}_2(y_i)$ (multiplication refers to $\otimes$), we see that a sufficient condition for the system to be linearly independent is to have $[x_i \otimes \mathcal{H}_2(y_i)]\ \neq\ I_\pi \equiv [x_\pi \otimes \mathcal{H}_2(y_\pi)]$. Next, we show that with overwhelming probability, the system of linear equations is indeed independent for all $i \in \{1, .., n\}$:

- Recall that the range of $\mathcal{H}_2$ is $\{G\}^*$ and that the order of $\{G\}^* = (l - 1)$.
- Therefore, $\exists\ v_i,\ v_\pi\ \in \mathbb{F}_l^*$ such that $\mathcal{H}_2(y_i) = v_i \otimes G$ and $\mathcal{H}_2(y_\pi) = v_\pi \otimes G$.
- We can then re-write the sufficient condition as $x_i v_i \neq x_\pi v_\pi \pmod{l}$.

- Note that given x_i, x_π, and v_π, there is at most one value of $v_i \in \mathbb{F}_l^*$ that satisfies $x_i v_i = x_\pi v_\pi \pmod{l}$. Otherwise, we would have $v_i, v_i' \in \mathbb{F}_l^*$, $v_i \neq v_i' \pmod{l}$, and $x_i v_i = x_\pi v_\pi = x_i v_i' \pmod{l}$. This would imply that $v_i \equiv v_i' \pmod{l}$, a contradiction.
- Noting that each v_i corresponds to a distinct $\mathcal{H}_2(y_i)$, we conclude that given x_i, x_π and $\mathcal{H}_2(y_\pi)$ there is at most one $\mathcal{H}_2(y_i)$ s.t. $[x_i \otimes \mathcal{H}_2(y_i)] = I_\pi \equiv [x_\pi \otimes \mathcal{H}_2(y_\pi)]$.
- Since $\mathcal{H}_2$ is a RO outputing random values, the probability of getting the right value of $\mathcal{H}_2(y_i)$ is $\leq \frac{1}{|\{G\}^*|} < \frac{1}{|\{G\}|} < \frac{1}{2^k}$ (negligible in k).

$\forall i \in \{1, .., n\},\ i \neq \pi$, we therefore conclude that with overwhelming probability we have $[x_i \otimes \mathcal{H}_2(y_i)] \neq I_\pi$. In other terms, we can be confident that the linear system of 2 equations has a unique solution. Hence, $\forall i \in \{1, .., n\},\ i \neq \pi$, we have $(r_i)_a = (r_i)_b$, and $(c_i)_a = (c_i)_b$.

Moreover, we have

$$\sum_{i=1}^{n}(c_i)_a = \mathcal{H}_1^*(m_a, (L_1')_a, ., (L_n')_a, (R_1')_a, ., (R_n')_a) \pmod{l} \textit{ (since } \sigma_a \textit{ is a valid forgery)}$$

$$= \rho^*_{\alpha(\beta)} \neq \rho^{\sim}_{\alpha(\beta)} \textit{ (by design of the forgery tuples)}$$

$$= \mathcal{H}_1^{\sim}(m_b, (L_1')_b, ., (L_n')_b, (R_1')_b, ., (R_n')_b) \pmod{l} = \sum_{i=1}^{n}(c_i)_b \textit{ (since } \sigma_b \textit{ is a valid forgery)}$$

Since $\sum_{i=1}^{n}(c_i)_a \neq \sum_{i=1}^{n}(c_i)_b$, we conclude that $\exists j \in \{1, .., n\}$ *s.t.* $(c_j)_a \neq (c_j)_b$. But $\forall i \in \{1, .., n\},\ i \neq \pi$, we showed earlier that with overwhelming probability we have $(c_i)_a = (c_i)_b$. We then conclude that with overwhelming probability $(c_\pi)_a \neq (c_\pi)_b$.

Going back to the system of 2 equations associated with $i = \pi$, we write:

$$(r_\pi)_a + (c_\pi)_a x_\pi = log_G(L_\pi') = (r_\pi)_b + (c_\pi)_b x_\pi$$

That means that we can solve for $x_\pi = \frac{(r_\pi)_b - (r_\pi)_a}{(c_\pi)_a - (c_\pi)_b} \pmod{l}$ in polynomial time, contradicting the intractability of DL on elliptic curve groups. We conclude that the signature scheme is exculpable and hence secure against $EFACM_{Ex_\pi}$ in the RO model.

6 Security analysis - Anonymity

In this section, we show that Cryptonotes' signature scheme satisfies the weaker anonymity definition #2 introduced in part 3 of this series. The reason it cannot satisfy the stronger definition #1 has to do with the inclusion of the key image or tag. Linkable signatures in general cannot satisfy anonymity definition #1. By introducing a tag that is fully determined by a ring member's private key, anyone with access to the signature can rule-out or confirm whether a member was the author if the member releases her private key. To see this, suppose that ring member $\delta \in \{1, .., n\}$ releases her private key x_δ, and let $\sigma \equiv (I, c_1, .., c_n, r_1, .., r_n)$ be a valid signature. Anyone can calculate $x_\delta \otimes \mathcal{H}_2(y_\delta)$ and compare it to I. If the 2 values differ, then the signature could not have originated from member δ who can now be ruled out. The identity of

the signer is then confined to the remaining non-compromised members. On the other hand, if the 2 values match, then one can confidently assume that member δ was the author. The confidence is derived from the exculpability property of the scheme that we demonstrated earlier (i.e., with overwhelming probability, no one can forge a signature with a key image or tag corresponding to a non-compromised signer). This shows that a linkable signature scheme is not exculpable in the *anonymity*-metric sense, although it can be exculpable in the *unforgeability*-metric sense.

More formally, we let $\mathcal{A}(\omega)$ be a PPT adversary with random tape ω that takes 4 inputs:

- Any message m.
- A ring L of the n public keys $\{y_1, ...y_n\}$ of the ring members. L includes the public key y_π of the actual signer.
- A list $\mathcal{D}_t \equiv \{\hat{x}_1, ...\hat{x}_t\}$ of compromised private keys of ring members ($0 \leq t \leq n$). Note that $\mathcal{D}_t$ can be empty. Also note that $\hat{x}_i$ may be different than x_i but we always have $\mathcal{D}_t \subseteq \{x_1, ..., x_n\}$
- A valid signature $\sigma_\pi(m, L)$ on message m, with ring L and actual signer private key x_π.

$\mathcal{A}(\omega)$ outputs an index corresponding to the ring member in L that it thinks is the actual signer. Definition # 2 mandates that for any polynomial in security parameter k $Q(k)$, we have:

$$\frac{1}{n-t} - \frac{1}{Q(k)} \leq P[\mathcal{A}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L) \textit{ is valid}] \leq \frac{1}{n-t} + \frac{1}{Q(k)}$$
$$\text{if } x_\pi \notin \mathcal{D}_t \text{ and } 0 \leq t < n-1.$$

$$P[\mathcal{A}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L) \textit{ is valid}] > 1 - \frac{1}{Q(k)}$$
$$\text{if } x_\pi \in \mathcal{D}_t \text{ or } t = n-1.$$

In the RO model, we allow $\mathcal{A}(\omega)$ to send a number of queries (polynomial in k) to RO $\mathcal{H}_1$ and RO $\mathcal{H}_T$. The probability of $\mathcal{A}$'s success is then computed over the distributions of $\omega, \mathcal{H}_1$ and $\mathcal{H}_T$. Making explicit the dependence on the ROs, definition # 2's condition becomes:

$$\frac{1}{n-t} - \frac{1}{Q(k)} \leq P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L) \textit{ is valid}] \leq \frac{1}{n-t} + \frac{1}{Q(k)}$$
$$\text{if } x_\pi \notin \mathcal{D}_t \text{ and } 0 \leq t < n-1.$$

$$P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L) \textit{ is valid}] > 1 - \frac{1}{Q(k)}$$
$$\text{if } x_\pi \in \mathcal{D}_t \text{ or } t = n-1.$$

In order to prove that anonymity holds in the above sense, we proceed by contradiction and rely on the intractability of another hard problem over cyclic groups known as the *Decisional Diffie Hellman* problem (*DDH* for short). A proof that relies on an intractable problem is referred to as a *conditional* proof. In what follows, we first introduce the DDH problem and then prove anonymity of the Cryptonote's scheme.

DDH formulation: Let G be a generator of a cyclic group of order l. DDH states that if α and β are uniformly and independently chosen in $\mathbb{Z}_l^*$, then the value of $G^{\alpha\beta}$ looks like a random element in the group. Intuitively, this means that the following 2 distributions of tuples are indistinguishable:

- $(G^\alpha, G^\beta, G^{\alpha\beta})$, where α and β are randomly and independently chosen in $\mathbb{Z}_l^*$.
- $(G^\alpha, G^\beta, G^\gamma)$, where α, β, and γ are randomly and independently chosen in $\mathbb{Z}_l^*$.

In other terms, we don't know of any PPT algorithm M that when given a tuple $(G^\alpha, G^\beta, G^\gamma)$ as input, can do better than random guessing as to whether $\gamma = \alpha\beta$ or not. Formally, we describe the DDH problem as follows:

1. Let $(\alpha_0, \beta_0, \gamma_0)$ be uniformly drawn from $\mathbb{Z}_l^*$
2. Let (α_1, β_1) be uniformly drawn from $\mathbb{Z}_l^*$, and let $\gamma_1 = \alpha_1\beta_1$
3. Let b be uniformly drawn from $\{0, 1\}$
4. Let $(\alpha, \beta, \gamma) \equiv (\alpha_b, \beta_b, \gamma_b)$

   We don't know any $M \in PPT(k)$ (probabilistic polynomial time in k) such that $P[\ M(G^\alpha, G^\beta, G^\gamma) = b\ ]\ = \frac{1}{2} + \epsilon(k)$, for ϵ non-negligible in k.

The previous formalization means that if we randomly decide whether a tuple that we send to M is definitely a DDH instance or not, there is no known PPT(k) M that can tell what was decided with probability better than random guessing. Concretely:

- We flip a coin and assign the value 0 or 1 to variable b.
- If $b = 0$, we feed M a random tuple $(\alpha_0, \beta_0, \gamma_0)$. There is a negligible probability that $\gamma_0 = \alpha_0\beta_0$, but most likely $\gamma_0 \neq \alpha_0\beta_0$ and $(\alpha_0, \beta_0, \gamma_0)$ is not a DDH instance.
- If $b = 1$, we feed M a tuple $(\alpha_1, \beta_1, \alpha_1\beta_1)$ that is a DDH instance.

The objective is for M to devise a mechanism to guess, depending on the input tuple, whether $b = 0$ or $b = 1$. The intractability of DDH means that there does not exist M in the set of PPT(k) that can outperform random guessing in that endeavour.

Note that if $(G^\alpha, G^\beta, G^\gamma)$ is a DDH instance, we write $(G, G^\alpha, G^\beta, G^\gamma)$ to make explicit the group generator G. Clearly, a DDH instance by definition satisfies $\gamma = \alpha\beta$ and so will satisfy $\alpha = log_G(G^\alpha) = log_{G^\beta}(G^\gamma)$. In general, if (G, a, b, c) is a tuple with elements chosen from a group with generator G, saying that (G, a, b, c) is DDH is equivalent to saying that $log_G a = log_b c$. We now prove the anonymity of Cryptonote's scheme.

Anonymity proof : We consider 3 separate cases.

- Case 1: $x_\pi \notin \mathcal{D}_t$ and $0 \leq t < n - 1$.

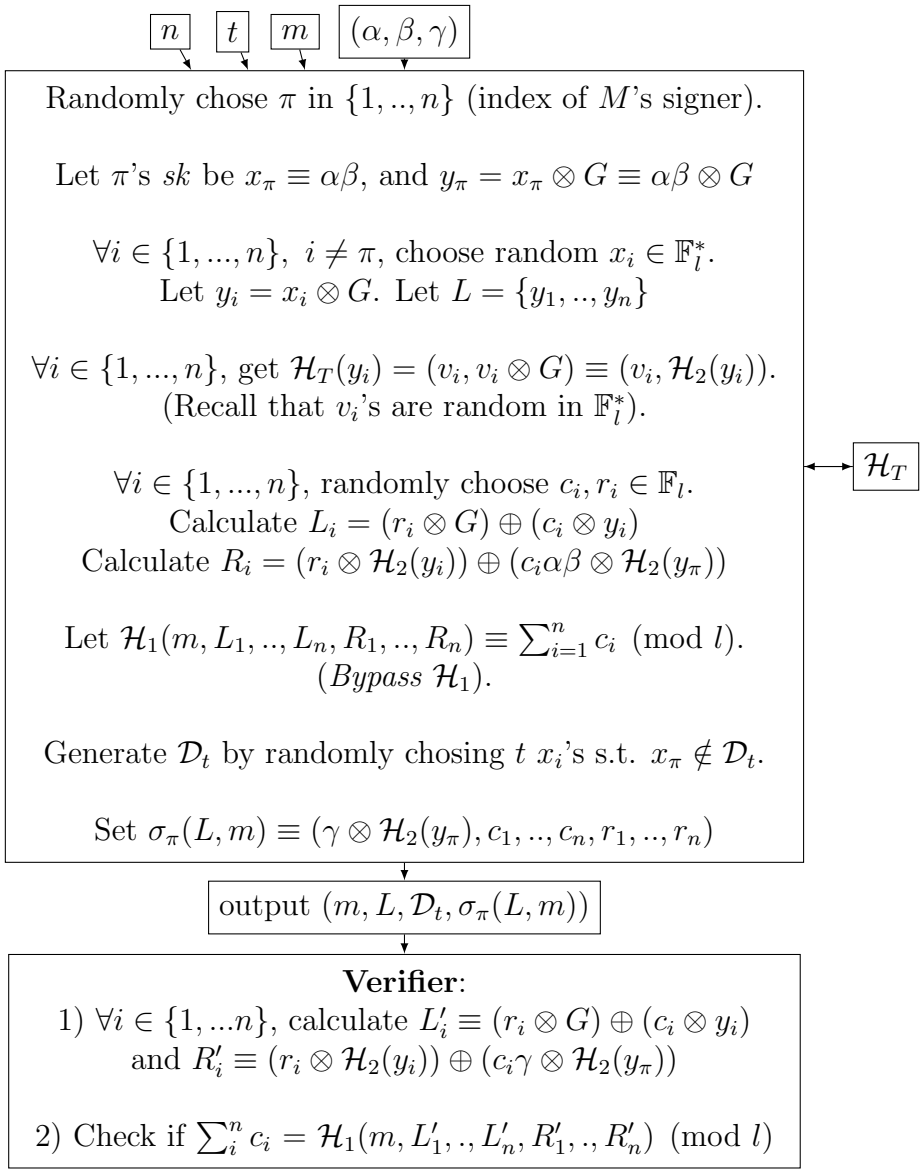
  Suppose that $\exists\ \mathcal{A}(\omega)$ in PPT(k) and $\epsilon(k)$ non-negligible in k such that

$$P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L) \; is \; valid] > \frac{1}{n-t} + \epsilon(k)$$
$$\text{if } x_\pi \notin \mathcal{D}_t \text{ and } 0 \leq t < n-1$$

Recall that since $x_\pi \notin \mathcal{D}_t$, one can automatically rule out all the compromised ring members as possible signers (the logic was described in the introductory paragraph of this section). One can then limit the guessing range of the identity of the signer to the uncompromised batch of $(n-t)$ remaining members.

We now build $M \in \text{PPT}(k)$ that colludes with $\mathcal{A}(\omega)$ to solve the DDH problem. M's input consists of 1) The tuple (α, β, γ) being tested for DDH, 2) A certain ring size n (randomly chosen), 3) A number $0 \leq t < n-1$ of compromised members (randomly chosen), and 4) A message m (randomly chosen).

M outputs a tuple consisting of 1) The message m, 2) A randomly generated ring L of size n, 3) A randomly chosen set $\mathcal{D}_t$ of t compromised secret keys, and 4) A not-necessarily valid signature $\sigma_\pi(L, m)$ assigned to ring member π s.t. $x_\pi \notin \mathcal{D}_t$.



M feeds its output $(m, L, \mathcal{D}_t, \sigma_\pi(L, m))$ to $\mathcal{A}(\omega)$. In order for $A(\omega)$ to use its advantage in guessing the signer's identity, it must be given a valid signature. For $\sigma_\pi(L, m)$ to be a valid signature, $(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ must be a DDH instance. Indeed, we have the following implications:

$$(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G) \text{ is DDH instance } \implies (\gamma = \alpha\beta)$$

$$\implies \forall i \in \{1, .., n\}$$
$$R_i = (r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i\alpha\beta \otimes \mathcal{H}_2(y_\pi)) = (r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i\gamma \otimes \mathcal{H}_2(y_\pi)) = R'_i$$

By design of M, we also have $\forall i \in \{1, .., n\}$, $L_i = L'_i$. We then conclude that

$$\textstyle\sum_{i=1}^{n} c_i \equiv \mathcal{H}_1(m, L_1, ., L_n, R_1, ., R_n) \pmod{l} = \mathcal{H}_1(m, L'_1, ., L'_n, R'_1, ., R'_n) \pmod{l}$$

$$\implies \sigma_\pi(L, m) \text{ is a valid signature.}$$

On the other hand, if $(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ is not a DDH instance, then $R_i \neq R'_i$ and with overwhelming probability

$$\textstyle\sum_{i=1}^{n} c_i \equiv \mathcal{H}_1(m, L_1, ., L_n, R_1, ., R_n) \pmod{l} \neq \mathcal{H}_1(m, L'_1, ., L'_n, R'_1, ., R'_n) \pmod{l}$$

and $\sigma_\pi(L, m)$ is not a valid signature.

Recall that $\mathcal{A}(\omega)$ can send queries to $\mathcal{H}_1$ and $\mathcal{H}_T$ during execution. It is important to enforce consistency between M and $\mathcal{A}(\omega)$'s query results obtained from RO $\mathcal{H}_1$ and RO $\mathcal{H}_T$ on the same input. There are no risks of faulty collisions in so far as $\mathcal{H}_T$ is concerned (by design of M). However, M bypasses RO $\mathcal{H}_1$ and conducts its own backpatching to $\mathcal{H}_1(m, L_1, .., L_n, R_1, .., R_n)$. If $\mathcal{A}(\omega)$ queries $\mathcal{H}_1$ on input $(m, L_1, .., L_n, R_1, .., R_n)$, then with overwhelming probability, it will conflict with M's backpatched value causing the execution to halt. The aforementioned collision must be avoided. In order to do so, we first calculate the probability of its occurence. We assume that during execution, $\mathcal{A}(\omega)$ can make a maximum of Q_1 queries to RO $\mathcal{H}_1$. Q_1 is assumed to be polynomial in the security parameter k, since the adversary is modeled as a PPT Turing machine.

$$P[Col] = P[\cup_{all\ (m,L_1,..,R_n)}\{(m, L_1, .., R_n) \textit{ appeared in } M \textit{ and in at least one of the } Q_1 \textit{ queries to RO } \mathcal{H}_1\}]$$

$$\leq P[\cup_{all\ L_1}\{L_1 \textit{ appeared in } M \textit{ and was part of at least one of the } Q_1 \textit{ queries to RO } \mathcal{H}_1\}]$$

$$\leq \textstyle\sum_{all\ L_1 \in \{G\}} P[\cup_{(j=1,..,Q_1)}\{L_1 \textit{ appeared in } M \textit{ and was part of at least the } j^{th} \textit{ query to RO } \mathcal{H}_1\}$$

$$\leq \textstyle\sum_{all\ L_1 \in \{G\}} \sum_{j=1}^{Q_1} P[L_1 \textit{ appeared in } M \textit{ and was part of at least the } j^{th} \textit{ query to RO } \mathcal{H}_1]$$

$$\leq \sum_{all\ L_1 \in \{G\}} \sum_{j=1}^{Q_1} \frac{1}{|\{G\}|^2} = |\{G\}| \times \frac{Q_1}{|\{G\}|^2} = \frac{Q_1}{|\{G\}|} < \frac{Q_1}{2^k}.$$

$$(since\ k < log_2(|\{G\}^*|) < log_2(|\{G\}|)\ by\ design).$$

and so we conclude that:

$$P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[(\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi) \cap \overline{Col} \mid \sigma_\pi(m, L)\ is\ valid] =$$

$$P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L)\ is\ valid] -$$
$$P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[(\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi) \cap Col \mid \sigma_\pi(m, L)\ is\ valid]$$

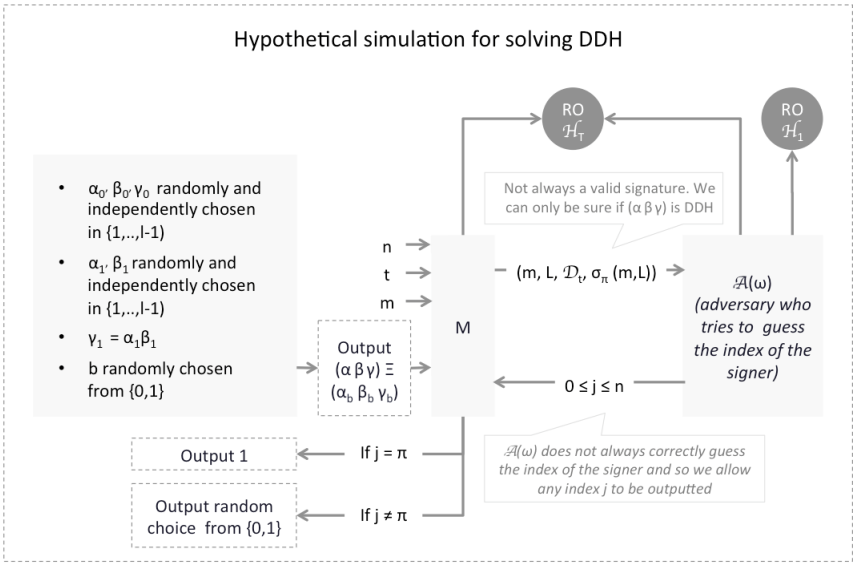
$$> P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L)\ is\ valid] - P[Col]$$

$$> P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L)\ is\ valid] - \frac{Q_1}{2^k}$$

$$> \frac{1}{n-t} + \nu(k)$$

whenever $x_\pi \notin \mathcal{D}_t$ and $0 \leq t < n-1$. Here, $\nu(k) \equiv \epsilon(k) - \frac{Q_1}{2^k}$ is non-negligibale in k

After execution, $\mathcal{A}(\omega)$ returns to M an integer $1 \leq j \leq n$. M then outputs 1 if $j = \pi$, or outputs 0/1 with equal probability otherwise. The following diagram summarizes the process:



Using the setting described above, we now calculate the probability of M guessing whether $(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ is DDH or not. In what follows we make use of the following notational simplifications:

- We refer to $M(\alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ simply as M.
- We refer to $\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L))$ simply as $\mathcal{A}(\omega)$.

We start by noticing that

$$P[M = b] = P[M = b|b = 1] \times P[b = 1] + P[M = b|b = 0] \times P[b = 0]$$

$$= \tfrac{1}{2} \times P[M = b|b = 1] + \tfrac{1}{2} \times P[M = b|b = 0]$$

1. Case $(b = 1)$: In this case, $(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ is a DDH instance and so as we saw earlier, $\sigma_\pi(m, L)$ will be a valid signature. $\mathcal{A}(\omega)$ would then use its hypothetical advantage to guess the index of the signer among the $(n - t)$ non-compromised ring members. We get:

$$P[M = b|b = 1] \geq P[(M = b) \cap (\overline{Col}) \ |b = 1] = P[(M = b) \cap (\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \ |b = 1] + \ P[(M = b) \cap (\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col})|b = 1]$$

$$= P[M = b \mid (b = 1), (\mathcal{A}(\omega) = \pi), (\overline{Col})] \times P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 1)] + P[M = b \mid (b = 1), (\mathcal{A}(\omega) \neq \pi), (\overline{Col})] \times P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 1)]$$

$$= 1 \times P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 1)] + \tfrac{1}{2} \times P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 1)]$$
(by design of M).

Since $\sigma_\pi(m, L)$ is a valid signature, we have:

$P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 1)] > \frac{1}{n-t} + \nu(k)$, for ν non-negligible in k. Let $P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 1)] = \frac{1}{n-t} + \zeta$ for some $\zeta \geq \nu(k)$. Hence $P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 1)] = 1 - \frac{1}{n-t} - \zeta$. We get:

$$P[M = b|b = 1] \geq 1 \times (\tfrac{1}{n-t} + \zeta) \ + \ \tfrac{1}{2} \times (1 - \tfrac{1}{n-t} - \zeta)$$

$$= \ \tfrac{1}{2} + \tfrac{1}{2(n-t)} + \tfrac{\zeta}{2} \ \geq \ \tfrac{1}{2} + \tfrac{1}{2(n-t)} + \tfrac{\nu(k)}{2}$$

2. Case $(b = 0)$: In this case, we do not know if $(G, \alpha \otimes G, \beta \otimes G, \gamma \otimes G)$ is a DDH instance or not, and hence can not be sure whether $\sigma_\pi(m, L)$ is a valid signature. Consequently, $\mathcal{A}(\omega)$ can no longer use its advantage in guessing the index of the signer, because this advantage works only when it is fed a valid signature. We get:

$$P[M = b|b = 0] \geq P[(M = b) \cap (\overline{Col}) \ |b = 0] = P[(M = b) \cap (\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \ |b = 0] + \ P[(M = b) \cap (\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \ |b = 0]$$

$$= P[M = b \mid (b = 0), (\mathcal{A}(\omega) = \pi), (\overline{Col})] \times P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 0)] + P[M = b \mid (b = 0), (\mathcal{A}(\omega) \neq \pi), (\overline{Col})] \times P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 0)]$$

$$= 0 \times P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 0)] + \tfrac{1}{2} \times P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 0)]$$
(by design of M).

and since $\mathcal{A}(\omega)$ can no longer use its advantage to guess the index of the signer, the best thing it can do is random guessing among non-compromised members. Hence $P[(\mathcal{A}(\omega) = \pi) \cap (\overline{Col}) \mid (b = 0)] = \frac{1}{n-t}$, and $P[(\mathcal{A}(\omega) \neq \pi) \cap (\overline{Col}) \mid (b = 0)] = 1 - \frac{1}{n-t}$. We get:

$$P[M = b|b = 0] \geq 0 \times (\frac{1}{n-t}) \ + \ \frac{1}{2} \times (1 - \frac{1}{n-t}) = \ \frac{1}{2} - \frac{1}{2(n-t)}$$

Putting it altogether, we conclude that:

$$P[M = b] \geq \frac{1}{2} \times (\frac{1}{2} + \frac{1}{2(n-t)} + \frac{\nu(k)}{2}) + \frac{1}{2} \times (\frac{1}{2} - \frac{1}{2(n-t)}) = \frac{1}{2} + \frac{\nu(k)}{4}$$

Since $\nu(k)$ is non-negligible in k, the above probability outperforms random guessing. This contradicts the intractability of DDH. Similarly, we can show $P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi] \mid \sigma_\pi(m, L)\ is\ valid]$ is also bounded from below. We finally conclude that for any polynomial $Q(k)$:

$$\frac{1}{n-t} - \frac{1}{Q(k)} \leq P_{\omega,\mathcal{H}_1,\mathcal{H}_T}[\mathcal{A}^{\mathcal{H}_1,\mathcal{H}_T}(\omega)(m, L, \mathcal{D}_t, \sigma_\pi(m, L)) = \pi \mid \sigma_\pi(m, L)\ is\ valid] \leq \frac{1}{n-t} + \frac{1}{Q(k)}, \text{ if } x_\pi \notin \mathcal{D}_t \text{ and } 0 \leq t < n-1.$$

- Case 2: $x_\pi \notin \mathcal{D}_t$ and $t = n - 1$.

  In this case, $\mathcal{A}(\omega)$ can check if I_π (the key-image or tag of $\sigma_\pi(m, L)$) matches any of the compromised tags $\hat{x}_i \otimes \mathcal{H}_2(\hat{x}_i \otimes G)$, for $i \in \{1, .., t = (n-1)\}$. With overwhelming probability, none of them will match since we proved that the scheme is exculpable and so no one can forge a signature with a tag of a non-compromised member. Proceeding by elimination, $\mathcal{A}(\omega)$ can then conclude that the signer is π.

- Case 3: $x_\pi \in \mathcal{D}_t$.

  In this case, $\mathcal{A}(\omega)$ can check which of the compromised tags $\hat{x}_i \otimes \mathcal{H}_2(\hat{x}_i \otimes G)$ $(i \in \{1, .., t\})$ matches I_π (the key-image or tag of $\sigma_\pi(m, L)$). Only one of them will match (due to exculpability), subsequently revealing the identity of the signer.

7 Security analysis - Linkability

In essence, the *linkability* property means that if a secret key is used to issue more than one signature, then the resulting signatures will be linked and flagged by $\mathcal{L}$ (the linkability algorithm). We claim that:

A signature scheme is linkable
$\Leftrightarrow$
$\forall n \in \{1, .., l-1\}, \forall L \equiv \{y_1, .., y_n\}$ a ring of n members, it is not possible to produce $(n+1)$ valid signatures with pairwise different key-images such that all of them get labeled *independent* by $\mathcal{L}$.

Proof of $\Leftarrow$: Consider the case $n = 1$ with $L \equiv \{y_1\}$. Then it is not possible to use y_1's secret key x_1 to produce 2 valid signatures such that they have different key-images and both are labeled *independent* by $\mathcal{L}$. In other words, the signature scheme is linkable.

Proof of $\Rightarrow$: We prove the contrapositive. Assume that $\exists$ a ring of n members $L \equiv \{y_1, .., y_n\}$ such that it can produce $(n+1)$ valid signatures with pairwise different key-images, and such that all of them get labeled *independent* by $\mathcal{L}$. This implies that $\exists i \in \{1, .., n\}$ such that the ring member with public key y_i produced at least 2 valid signatures with different key-images, both labeled independent by $\mathcal{L}$. This means that the scheme is not linkable.

To prove that Cryptonote's scheme is linkable we follow a *reductio ad absurdum* approach:

- Assume that the scheme is not linkable.
- The equivalence above would imply that $\exists L \equiv \{y_1, .., y_n\}$ such that it can produce $(n+1)$ valid signatures with pairwise different key-images (i.e.,$\forall i, j \in \{1, .., n\}, i \neq j \Rightarrow (I_i \equiv x_i \otimes \mathcal{H}_2(y_i)) \neq (I_j \equiv x_j \otimes \mathcal{H}_2(y_j))$), and such that all of them get labeled *independent* by $\mathcal{L}$.
- This means that there must exist a signature (from the set of $(n+1)$ valid signatures) with key-image I_δ such that $\forall i \in \{1, .., n\},\ I_\delta \neq I_i \equiv x_i \otimes \mathcal{H}_2(y_i)$. Denote this signature by $\sigma_\delta \equiv (I_\delta, c_1, .., c_n, r_1, .., r_n)$.
- When verifying the validity of σ_δ, $\mathcal{V}$ will first compute the following for all $i \in \{1, .., n\}$:

  $\{\ L'_i = (r_i \otimes G) \oplus (c_i \otimes y_i)$

  $\{\ R'_i = (r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i \otimes I_\delta)$
- $\forall i \in \{1, .., n\}$, the system of 2 equations above can be equivalently written as:

  $\{\ r_i + c_i x_i = log_G(L'_i)$

  $\{\ r_i \otimes \mathcal{H}_2(y_i)) \oplus (c_i \otimes I_\delta) = R'_i$

  For a given L'_i, R'_i, and $i \in \{1, .., n\}$, this constitutes a system of 2 equations in variables r_i and c_i.
- Since $\forall i \in \{1, .., n\},\ I_\delta \neq x_i \otimes \mathcal{H}_2(y_i)$, the system of 2 equations corresponding to each i is independent and admits a unique solution (r^*_i, c^*_i) for any given L'_i, and R'_i. That means that for given L'_i, and R'_i, the value $c^* = \sum_{i=1}^{n} c^*_i$ is well defined.
- By virtue of being a valid signature, σ_δ must satisfy $\mathcal{V}$'s verification equation. More specifically, it must be that $\mathcal{H}_1(m, L'_1, ..L'_n, R'_1, .., R'_n) \pmod l = c^*$. But RO $\mathcal{H}_1$ is random by definition. The probability that it outputs a specific value is eqal to $\frac{1}{q}$ (recall that the range of $\mathcal{H}_1 = \mathbb{F}_q$). And since by design we have $2^k < l - 1 < l < q$, we conclude that the probability that $\mathcal{H}_1(m, L'_1, ..L'_n, R'_1, .., R'_n) \pmod l = c^*$ is upper-bounded by $\frac{1}{2^k}$ and is hence negligible. In other terms, the probability that σ_δ is a valid signature is negligible.

We can then conclude that with overwhelming probability, the ring $L \equiv \{y_1, .., y_n\}$ can not produce $(n+1)$ valid signatures with pairwise different key-images and such that all of them get labeled *independent* by $\mathcal{L}$. Cryptonote's scheme is hence linkable.

References

[1] E. Fujisaki and K. Suzuki. Traceable ring signatures. *Public Key Cryptography*, pages 181–200, 2007.

[2] N. Van Saberhagen. Cryptonote 2.0. https://cryptonote.org/whitepaper.pdf, 2013.